WWW.HACKERJOURNAL.IT
HACKER JOURNAL
N° 198
2€
NO PUBBLICITÀ
SOLO INFORMAZIONI E ARTICOLI
SPECIALE OSSEC: COME SCOVARE L'INTRUSO
AIRCRACK ALL'ASSALTO DELLE CHIAVI WEP E WPA
VERY SECURE FTP DAEMON
FTP SICURO
PERSONAGGI
› STALLMAN IL PALADINO DEL FREE SOFTWARE
COMPUTER
› EXFAT, UN FILE SYSTEM POCO NOTO
SDK 3.2
› PROGRAMMARE PER IPAD
QUATTORD. ANNO 10 - N° 198 - 1 APRILE/14 APRILE 2010 - € 2,00
WLF PUBLISHING

EDITORIALE

ASPETTATIVE, CONFERME E LIETE SORPRESE

L'uscita di una rivista genera molta aspettativa sia per chi la legge, sia per chi la realizza.
E' una sorta di appuntamento al buio. Nessuno sa con certezza se le idee della redazione incontreranno il favore dei lettori. Nessuno può prevedere con ragionevole approssimazione se ci sarà un "felice matrimonio" tra contenuti e gusti dei lettori.
Negli ultimi numeri abbiamo cercato di capire come Hacker Journal dovrebbe evolversi per arrivare, magari tra qualche anno, a festeggiare il numero 300 in piena salute (del duecentenario imminente ne parleremo nei prossimi numeri).
Molte indicazioni, sia nella posta che sul forum, sembrano apprezzare una sorta di svolta tecnica che è avvenuta soprattutto a partire dal numero 196.
C'è, da parte dei lettori, molta curiosità, voglia di approfondire ma, soprattutto, competenza.
Ci siamo accorti, leggendo i contributi inviati al "Laboratorio" di HJ, come lo skill di alcuni utenti della rivista sia davvero notevole. Paradossalmente potrebbero davvero rubarci il lavoro e venire qui in redazione a fare la rivista e non sono altrettanto sicuro che noi della redazione saremmo altrettanto attenti e partecipi come lettori. Paradossi a parte, proseguiamo nella direzione che molti di voi sembrano indicarci.
Come leggerete nella rubrica della posta "incombe" un serio e articolato corso di programmazione in C. Che ne pensate?

laboratorio@hackerjournal.it
Questo indirizzo è stato creato per inviare articoli, codice, spunti e idee. E' quindi proprio una sorta di "incubatore di idee".

posta@hackerjournal.it
E' l'account creato per l'omonima rubrica che è ricomparsa nelle pagine della rivista. A questo indirizzo dovete inviare tutte le mail che volete vengano pubblicate su HJ.

redazione@hackerjournal.it
Questo è l'indirizzo canonico. Quello con cui potete avere un filo diretto, sempre, con la redazione, per qualsiasi motivo che non rientri nelle due precedenti categorie di posta.

Sommario

Anno 10 - N.198
1 aprile / 14 aprile 2010

Editore (sede legale)
WLF Publishing S.r.l.
Socio Unico Medi & Son S.r.l.
via Donatello 71 - 00196 Roma
Fax 063214606

Realizzazione editoriale
Progetti e promozioni Srl
redazione@progettiepromozioni.com

Printing
Grafiche Mazzucchelli S.p.a - Seriate (BG)

Distributore
M-DIS Distributore SPA
via Cazzaniga 2 - 20123 Milano

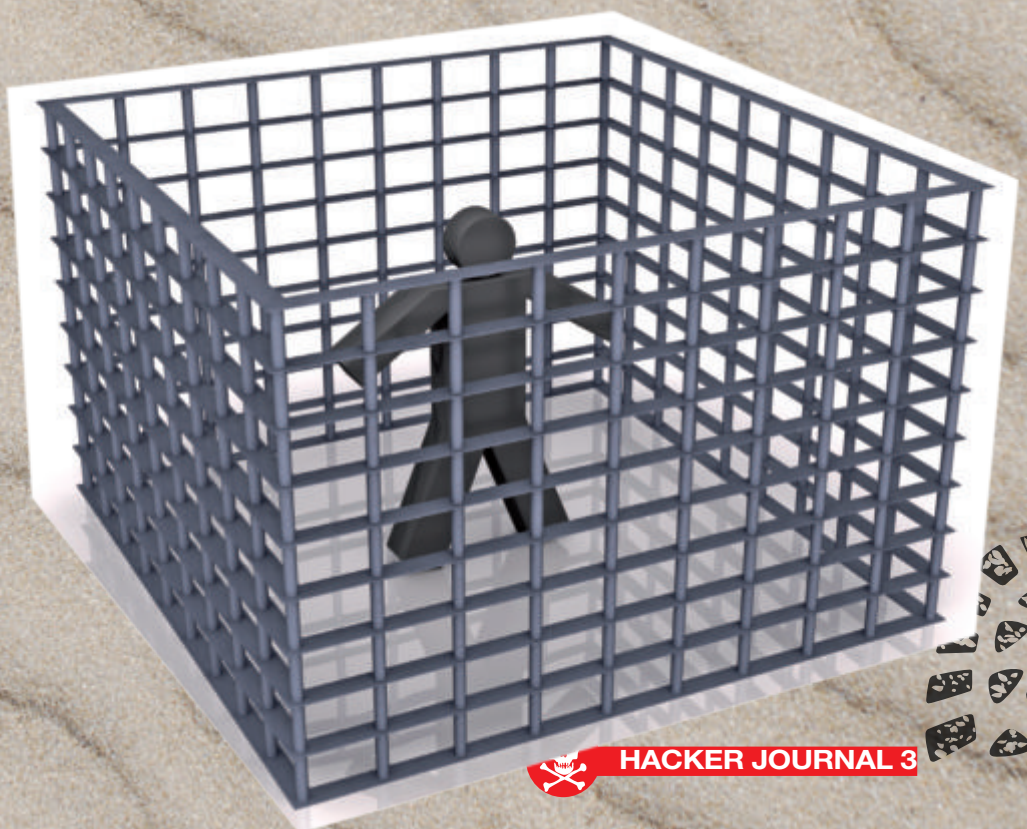
Hacker Journal
Pubblicazione quattordicinale registrata al Tribunale di Milano il 27/10/03 con il numero 601.
Una copia: 2,00 euro

Direttore Responsabile
Teresa Carsaniga
redazione@hackerjournal.it

NeWS

Un mondo sempre più CHIUSO

Se il mondo intero, quello reale, sembra sempre più volto ad una globalizzazione estrema, con libera circolazione di risorse e persone, il mondo informatico presenta piccoli preoccupanti segni di involuzione.
Qualche anno fa si pensava che il software open source potesse sostituire i costosi software commerciali e rappresentare una sorta di modello condiviso e accessibile a tutti.
Nel 2010 la situazione è rimasta pressoché immutata. Non c'è stata la grande affermazione dell'open source e, anzi, si stanno diffondendo modelli di business, informatico, sempre più chiusi.
Qualche esempio?
Gli store legati alla telefonia, come ad esempio Apple Store, propongono un modello di distribuzione del software chiuso, anzi blindato. Lo sviluppatore firma il codice e l'applicazione caricata sul sito viene impacchettata coi codici sorgente "secretati", inoltre, per scaricarla, bisogna avere un particolare terminale, come l'iPhone per l'Apple Store, altrimenti il meccanismo di download non si può perfezionare.
E' un sistema rigidamente chiuso su se stesso, manco fosse la prigione di massima sicurezza di Alcatraz (peraltro ormai dismessa tanto per significare che i tempi cambiano un po' per tutto e tutti).
Però curiosamente sono meccanismi di distribuzione che funzionano per la telefonia, tant'è che all'Apple Store sono seguiti l'Ovi Store, lo Store di Android, il Market Mobile di Windows ecc…
Ora, poi, con l'imminente avvento dell'iPad e l'apertura dell'iBookStore, sembra che questa filosofia di distribuzione possa allargarsi anche ad apparecchi diversi dagli smartphone, finendo per condizionare anche il mondo dei computer. Si tratta di previsioni pessimistiche e per nulla permeate di realismo?
Forse.
Certo è che Richard Stallman, presente all'interno della rivista, che da anni si batte per il libero pensiero e la libera distribuzione del software, invecchia professando la sua affascinante e condivisibile filosofia in giro per il mondo. Lo stesso mondo che sembra cambiare in direzione opposta, scegliendo modelli decisamente più remunerativi.
Forse l'utopia del software accessibile a tutti rimane, appunto, un'utopia, bella, affascinante ma irrealizzabile.
Però ci potrebbe essere un ragionevole compromesso che le leggi di mercato tendono continuamente a scoraggiare.

NeWS

IN PILLOLE

OPENSSL È VULNERABILE

★ E' stata rilevata una vulnerabilità in OpenSSL che, però, può essere sfruttata solo con un intervento hardware, e OpenSSL è già al lavoro per realizzare una patch.
La scoperta si deve ad Andrea Pellegrini, Valeria Bertacco e Todd Austin, ricercatori dell'Università del Michigan. Secondo il loro test bastano 100 ore di calcolo computazionale per ottenere la chiave da 1.024 bit, ma l'attacco deve essere portato a livello hardware agendo sul voltaggio del chip sotto attacco.

SMANTELLATA LA BOTNET MARIPOSA

★ Dopo più di un anno di attività, le autorità spagnole hanno smantellato la botnet Mariposa, arrestando i tre responsabili che durante la loro attività hanno infettato più di 13 milioni di computer. Tra i computer infetti figurano quelli di una buona metà delle 1.000 aziende della classifica stilata da Fortune, e una quarantina tra le maggiori società bancarie mondiali. Questo è un dato estremamente preoccupante che deve fare riflettere sull'importanza di una corretta politica sulla sicurezza, specie a livello aziendale.

BACO PER LA playstation 3

Panico per gli utenti della Playstation per un problema di programmazione legato all'orologio interno che si è verificato con il passaggio del calendario da 28/02/2010 a 1/03/210. Praticamente accendendo alcune console FAT (Le slim sono risultate esenti) venivano riportate, come calendario, al 1999 o 2000 perdendo salvataggi e manifestando altri malfunzionamenti (impossibilità collegarsi online con la rete PlayStation Network e di giocare diversi titoli anche in modalità off-line). L'errore "dichiarato" è l'8005010F. Perché sia potuto avvenire lo sanno solo i programmatori di Sony (non avendo altri accesso al sistema). Peraltro il bug è stato superato col passaggio al 2 Marzo, il giorno successivo. Non c'è stato bisogno di intervento da parte dei tecnici Sony, come era peraltro preventivabile, la PS3 ha fatto tutto da sé.

SPYBOT.AKB: IL WORM CHE SFRUTTA LE RETI P2P E LE E.MAIL

I laboratori di Panda Security hanno scoperto un nuovo worm: Skybot.AKB che si diffonde attraverso le reti P2P e tramite e.mail.
La novità è la tecnica utilizzata per ingannare gli utenti, diffondendosi come un falso invito per usare i social network come Twitter (http://www.flickr.com/photos/panda_security/4367549638/) e Hi5 o con un messaggio di posta elettronica proveniente da Google nel qual si offre lavoro. Inoltre, il codice maligno si fa passare come estensione di Firefox security (http://www.flickr.com/photos/panda_security/4366803381/).

Questi sono alcuni esempi dei testi che compaiono:

- Jessica would like to be your friend on hi5!
- You have received A Hallmark E-Card!
- Shipping update for your Amazon.com order 254-71546325-658732
- Thank you from Google!
- Your friend invited you to twitter!

Una volta installato nel Pc, quando l'utente avvia una ricerca con le seguenti parole chiave, viene diretto verso un contenuto totalmente diverso da quello atteso:
A: Airlines, Amazon,Antivir, Antivirus.
B: Baseball, Books.
C: Casino, Chrome, Cialis, Cigarettes, Comcast, Craigslist, Credit.
D: Dating, Design, Doctor. E: Explorer
F: Fashion, Finance, Firefox, Flifhts, Flower, Football
G: Gambling, Gifts, Graphic.
H: Health, Hotel.
I: Insurance, Iphone.
L: Loans.
M: Medical, Military, Mobile, Money, Mortgage, Movie, Music, Myspace.
O: Opera.
P: Pharma, Pocker. S: School, Software, Sport, Spybot, Spyware.
T: Trading, Tramadol, Travel, Twitter.
V: Verizon, Video, Virus, Vocations. W: Wallpaper, Weather.

Il worm porta a termine delle azioni volte a ridurre l'efficacia della sicurezza della macchina stessa, come l'aggiunta alla lista delle applicazioni autorizzate dal firewall di Windows, disattivare il servizio dei report di errori di Windows o il servizio utenti per il controllo accessi (UAC).

LE PREVISIONI METEO SONO "PERICOLOSE"

Cercando di aumentare la consapevolezza circa i rischi per la sicurezza che comporta l'utilizzo di applicazioni di terze parti per smartphone, una coppia di ricercatori ha utilizzato un'innocua applicazione meteo per comandare circa 8.000 iPhone e dispositivi Android in una botnet di telefonia mobile.
Il progetto di ricerca è stato presentato alla conferenza RSA di quest'anno per mostrare come innocue applicazioni per smartphone possano raccogliere informazioni sensibili degli utenti, comprese le coordinate GPS e i numeri di telefono.
Il progetto è il frutto di una collaborazione tra i ricercatori Derek Brown e Daniel Tijerina TippingPoint con il sito Weather Underground che fornisce informazioni sulle previsioni meteo per i suoi utenti.
L'applicazione, chiamata WeatherFirst, per poter fornire informazioni e previsioni sul meteo accede direttamente alle coordinate Gps dell'utente, incrociandole con il suo numero di telefono.
I due ricercatori hanno creato anche una versione "malevola" di WeatherFirst identica alla prima, ma che riesce anche a rubare le informazioni da smartphone sbloccati, quindi che non scaricano le app dagli store abituali come l'Apple Store dove il processo di sicurezza è rigoroso e include la firma del codice per potere distribuire la propria applicazione.

Falla critica all'opera in... Opera

E' stato rilevato un difetto nel Browser Opera che consentirebbe l'esecuzione di codice arbitrario a distanza.
La vulnerabilità è causata da un errore durante l'elaborazione delle risposte HTTP con Content-Length malformato.
Questa vulnerabilità può essere sfruttata per impostare un attacco di tipo buffer overflow.
In assenza di una patch, gli utenti di Opera sono invitati ad evitare la navigazione a siti Web non attendibili oppure passare a un browser alternativo.

LA CLASSIFICA DEI MALWARE

I Kaspersky Lab. Hanno pubblicato la lista dei programmi malware, adware e programmi potenzialmente pericolosi che sono stati individuati e neutralizzati dalla scansione online nel mese di Febbraio 2010.
A lato si può leggere la speciale classifica.

Le prime 5 posizioni non sono mutate rispetto al mese scorso, e a giudicare dal numero di infezioni, l'epidemia di Kido ha leggermente perso vigore.

Exploit.JS.Aurora.a, un exploit molto attivo che approfitta delle vulnerabilità di diversi software, è entrato direttamente in settima posizione.

Tra le new entry, una coppia di programmi adware FunWeb.q in ventesima posizione, è un perfetto esempio di adware: è una toolbar per i browser più diffusi, e consente agli utenti un accesso facilitato alle risorse di alcuni siti web (di solito quelli a contenuto multimediale). FunWeb.q modifica inoltre le pagine che l'utente visita in modo da mostrare i banner pubblicitari.

Le cose si fanno più complicate nel caso di un non-virus: AdWare.Win32.RK.aw, che occupa la tredicesima posizione. Si tratta dell'applicazione RelevantKnowledge che si diffonde e installa assieme ad altri software. Nelle norme di privacy stabilite dalla società produttrice (http://www.relevantknowledge.com/RKPrivacy.aspx) l'utente viene informato che il programma traccerà virtualmente tutta la sua attività, soprattutto su Internet, raccogliendo automaticamente le informazioni personali del malcapitato, e salvandole sui propri server. Prosegue dicendo che tutti questi dati raccolti verranno usati con il solo scopo di "aiutare a creare il futuro di Internet" e che tutti i dati saranno ben protetti. Sta poi all'utente decidere se credere o meno a quanto riportato.

NOME	INFEZIONI
1 Net-Worm.Win32.Kido.ir	274729
2 Virus.Win32.Sality.aa	179218
3 Net-Worm.Win32.Kido.ih	163467
4 Net-Worm.Win32.Kido.iq	121130
5 Worm.Win32.FlyStudio.cu	85345
6 Trojan-Downloader.Win32.VB.eql	56998
7 Exploit.JS.Aurora.a	49090
8 Worm.Win32.AutoIt.tc	48418
9 Virus.Win32.Virut.ce	47842
10 Packed.Win32.Krap.I	47375
11 Trojan-Downloader.WMA.GetCodec.s	43295
12 Virus.Win32.Induc.a	40257
13 Non virus:AdWare.Win32.RK.aw	39608
14 Non virus:AdWare.Win32.Boran.z	39404
15 Worm.Win32.Mabezat.b	38905
16 Trojan.JS.Agent.bau	34842
17 Packed.Win32.Black.a	32439
18 Trojan-Dropper.Win32.Flystud.yo	32268
19 Worm.Win32.AutoRun.dui	32077
20 Non virus:AdWare.Win32.FunWeb.q	30942

:: POSTA ::

CORSO DI PROGRAMMAZIONE IN C

Faccio presente le statistiche sul forum riguardo il corso di programmazione in C: 18 voti a favore, credo siano sufficienti no? Se si farà come verrà impostato? gli utenti potranno partecipare? A tal proposito lascio il link del topic http://www.hackerjournal.it/HJ/forum/viewtopic.php?f=10&t=201.

Lorenzo G.

In effetti, incrociando le riposte sul forum e quelle pervenute via mail, ci sembra di capire che il corso di programmazione in C goda di una certa aspettativa, motivo per cui abbiamo deciso di incaricare due nostri validi collaboratori, uno per inciso è Giovanni Federico che leggete abitualmente sul forum e sulla rivista, l'altro è Fabio `BlackLight` Manganello, di pensare ad una possibile impostazione.
Il problema di un corso di programmazione, in generale, e di un corso di programmazione in C, in particolare, è quello di cercare di condensare in poco spazio una grande quantità di argomenti, cercando di non essere superficiali, ma approfondendo tutti gli aspetti salienti.
La suddivisione dovrebbe essere in 7 parti

•Introduzione al linguaggio, Tipi di dato, Operatori, Strutture di controllo (condizionali ed iterative).

•Programmazione funzionale: dichiarazione, implementazione ed uso di funzioni.

•Vettori: dichiarazione, accesso ed inizializzazione; algoritmi di ricerca lineare e dicotomica, ordinamento e vettori multidimensionali.

•I puntatori e la gestione della memoria (allocazione e rilascio).

•Unioni e nuovi tipi, Strutture dati: liste e stack (modello esemplificativo, push/pop con liste doppiamente concatenate).

•Stringhe: modifica, ricerca, confronto e copia. Operazioni sui file: lettura, scrittura e file con liste collegate.

•Quiz di autovalutazione (domande su HJ X e risposte sul sito).

Abbiamo in programma di partire sul numero 200. Ci sembra sia di buon auspicio, sia un buon modo per celebrare duecento numeri e otto anni di pubblicazione della rivista. Lunga vita ad HJ.

ABBONAMENTO

Sono molto contento della vostra rivista, tocca argomenti molto interessanti e utili.
Colgo l'occasione per chiedere se sarà possibile prima o poi sottoscrivere un abbonamento, non ho trovato sul sito la possibilità di farlo.

Luca

Abbiamo pubblicato questa mail come rappresentativa delle decine che riceviamo in merito alla richiesta di abbonamento. Purtroppo la possibilità di abbonarsi non è mai rientrata nelle logiche editoriali di Hacker Journal e probabilmente non vi rientrerà neanche prossimamente, quindi l'unico posto per reperire la rivista rimane l'edicola. Se vi saranno dei cambiamenti, naturalmente, ve lo faremo sapere attraverso le pagine del giornale, ma al momento l'ipotesi non è prevista.

SOTTO ATTACCO

di Federico H94F
carbone18@gmail.comt

AirCrack

TESTARE LA PROPRIA RETE WIRELESS

WIRELESS

AirCrack è uno dei più diffusi tool per "ascoltare" e monitorare il traffico su una rete wireless, ma può essere usato anche per altri scopi...

Qual è il grado di sicurezza di una rete Wi-Fi? Secondo i più esperti in materia basta circa un minuto per scovare la chiave di accesso di una rete wireless protetta dal noto metodo WEP. Poiché non sono poi così tante le modalità di protezione di una rete internet (Wireless) è più facile per gli "scrocconi" accedervi o perlomeno portare degli attacchi mirati..
Per chi non ne fosse a conoscenza i metodi di protezione di una rete Wi-Fi sono soltanto due: WEP e WPA. La modalità di protezione che al momento dà maggiore sicurezza è la WPA anche se quella largamente più utilizzata è, invece, la WEP.

WEP E WPA

Prima di illustrare i programmi usati maggiormente approfondiamo il significato di queste due sigle WEP e WPA:
• WEP: corrisponde a Wired Equivalent Privacy, fa parte dello standard IEEE 802.11, specifico per rendere "sicuro" l'utilizzo delle reti Wi-Fi.Poichè questa chiave di protezione con il tempo si è rivelata ormai scarsa in termini di protezione è stata progettata la chiave WPA presente dal 2003. Questa usa l'algoritmo di cifratura stream RC4 per la sicurezza e CRC-32 per l'integrità dei dati.
• WPA: che corrisponde a Wi-Fi Protected Access è il protocollo realizzato per colmare le falle del precedente, più sicuro ma con ancora diversi difetti. Il protocollo TKIP cambia dinamicamente la chiave in uso e la combina con un vettore di inizializzazione (IVS) di dimensione doppia rispetto al WEP (in modo da rendere vani gli attacchi simili a quelli previsti per il WEP) e può essere implementato nelle schede di interfaccia wireless pre-WPA, che cominciarono ad essere distribuite nel 1999, attraverso un aggiornamento del firmware.
• WPA2: ancora poco usata, è la nuova chiave che dovrebbe sostituire la "vecchia" WPA.
Per "rubare" una chiave WEP non c'è bisogno di essere un esperto informatico, bastano alcuni strumenti e qualche programmino in grado di intercettare i pacchetti che vengono trasmessi e decriptarli, naturalmente più pacchetti vengono captati più sono le possibilità di arrivare all'obiettivo.

AIRCRACK

Tra i programmi più usati abbiamo AirCrack scaricabile da questo indirizzo: http://downloads.phpnuke.org/it/download-item-view-a-b-m-g-x.htm
Aircrack-ng è un pacchetto che permette di recuperare password dalle reti 802.11 WEP, WPA e WPA2-PSK. Il programma funziona in modo semplice, acquisisce i pacchetti che circolano sulla rete controllata ed esegue le fasi tradizionali di recupero: brute force, dictionary ecc.
Inoltre, include gli strumenti necessari per ripristinare velocemente l'accesso a una rete senza fili.

I "FERRI DEL MESTIERE"

Eseguiremo l'applicazione dal prompt di DOS: per poter eseguire il programma anche se non siamo nella directory che lo contiene, cliccate col tasto destro del mouse su "Risorse del computer, Proprietà, Avanzate, Variabili

d'ambiente, Modifica" portatevi alla fine della linea su cui sono già scritte altre variabili d'ambiente, inserite un punto e virgola (";") e scrivete il percorso in cui si trova il programma.
Avrete anche bisogno di installare dei nuovi driver per il vostro adattatore di rete: i driver originali non sono stati pensati per fare cose simili (per trovare dei driver che possano fare al caso vostro visitate il sito WildPacket online su www.wildpackets.com).
Per installare i nuovi driver: "clic destro su Risorse del Computer, Proprietà, Gestione Periferiche, clic destro sul vostro adattatore di rete, Proprietà, Driver, Aggiorna Driver, Installa da un elenco o percorso specifico". Scegliete il percorso in cui avete scaricato i driver. Assicuratevi, infine, che il vostro adattatore di rete sia ora compatibile col tutto

Il primo passo è ovviamente quello di trovare una rete wireless.
Digitate "Airodump" nel prompt del DOS (Start, esegui, cmd).
Vi comparirà una finestra contenente le schede di rete trovate sulla vostra macchina. Notate che accanto al nome delle schede di rete è presente un numero identificativo. Ad esempio:

14 NETGEAR WG511T 54 Mbps Wireless PC Card
22 NETGEAR WAG511 802.11a/b/g Dual Band Wireless PC Card

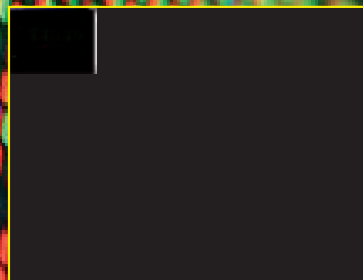
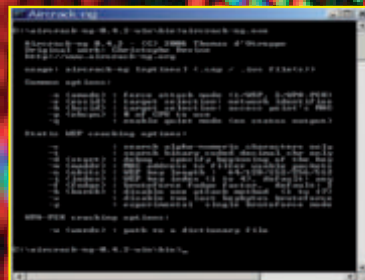
In questo caso digitate 22, il numero identificativo della scheda che ci interessa utilizzare. Ora vi viene chiesto di indicare il chipset utilizzato dal vostro adattatore di rete. Ad esempio:

Interface types: 'o' = HermesI/Realtek
'a' = Aironet/Atheros

Quindi vi sarà richiesto di inserire il numero del canale da controllare (sniffing). Per l'Europa il 14. Se volete fare una scansione di tutti i canali utilizzate lo zero.

In seguito il programma chiederà di digitare il nome da dare al file che verrà creato a partire dalla scansione del canale. Digitate il nome che vi pare, ad esempio "WEP01".

A questo punto Aircrack vi chiederà se salvare gli interi pacchetti catturati o soltanto gli IV. Per craccare una chiave WEP vi basta salvare semplicemente gli IV (il che vi farà risparmiare diverso spazio sull'hard disk, quindi digitate "y").

Adesso vedrete una schermata simile:

AirCrack-ng può essere scaricato all'indirizzo http://downloads.phpnuke.org/it/download-item-view-a-b-m-g-x.htm.

BSSID = l'indirizzo MAC dell'Access Point.

PWR = indica la forza del segnale che si sta ricevendo.

BEACONS = sono pacchetti "in chiaro" che l'Access Point trasmette sostanzialmente per dire "sono un access point, collegati a me".

DATA = è quello che ci interessa: sono gli IV che Aircrack utilizzerà per trovare la password WEP.

ENC = il tipo di incapsulamento: WEP, WPA, OPEN…

ESSID = Il nome della rete Wireless. L'SSID è una sorta di identificativo della rete. Se ad esempio l'Access Point ha come SSID il nome "pippo" allora le schede wireless che si vogliono connettere devono impostare a loro volta come SSID "pippo". Nella seconda parte dell'immagine, sopra, vediamo i vari client che stanno "dialogando" con l'Access Point", più esattamente vediamo i vari indirizzi MAC dei client. Quest'informazione può risultare utile in seguito quindi bisogna ricordarsi tutto. Ciò significa che anche se possediamo la chiave WEP non possiamo accedere all'Access Point a meno che l'indirizzo MAC del nostro adattatore di rete non sia stato impostato nel filtro dell'Access Point. Aicrack continuerà a collezionare IV finché non lo fermate. Più IV scaricate e più possibilità avete di decifrare la chiave. Aircrack dovrebbe trovare la chiave in pochi secondi. Per una chiave da 104 bit collezionate circa 2.000.000 di IV: a volte ne bastano molti meno, altre volte purtroppo dovrete scaricarne di più. Quando sarete soddisfatti del numero di IV collezionati premete "CTRL + C" per fermare il programma.

Scrivendo "Aircrack-ng" nel prompt vi verrà mostrata la lista dei parametri che è possibile utilizzare. Per esempio, se abbiamo scaricato intorno ai 400.000 IV, in genere sufficienti per scovare una chiave WEP da 40 bit, allora digitiamo il comando "aircrack-ng -n 64 WEP1.ivs". Con il parametro "-n 64" diciamo al programma che la chiave ha una lunghezza massima di 64 bit e di non provare quindi oltre. Se riusciremo nell'intento, il programma ci restituirà un messaggio di "KEY FOUND" seguito dal nome della chiave. Ora che avete la chiave utilizzatela proprio come se doveste connettervi ad una vostra rete "domestica". Se l'SSID dell'Access Point è abilitato seguite questo passo.

Premete su Start, Connetti a, Connessioni di rete senza fili, Visualizza reti senza fili disponibili. Se l'SSID è abilitato, questo vi appare nella finestra delle connessioni disponibili. Fate doppio clic sull'icona della connessione ed inserite la password che avete trovato in precedenza: missione compiuta.

Massimiliano Rinaldi
redazione@hackerjournal.it

SDK 3.2 programmare per iPad

PROGRAMMARE

IL NUOVO SOFTWARE DEVELOPMENT KIT PER PROGRAMMARE APPLICAZIONI IPAD È GIÀ DISPONIBILE...

Sognate di diventare ricchi con un'applicazione? A qualcuno è successo sviluppando per iPhone, specie all'inizio, ora il nuovo iPad potrebbe proporre opportunità non indifferenti ai nuovi, "vecchi" sviluppatori.
Nuovi perché si apre un mercato per tutti coloro che vogliono cominciare a sviluppare col Software Development Kit proprio su iPad, vecchi perché gli strumenti di sviluppo sono assai simili a quelli che molti utenti utilizzano già per sviluppare applicazioni per iPhone.
Sul sito Apple Developer Connection, nella sezione iPhone Dev Center è stato reso disponibile il nuovo kit di sviluppo, Software Development Kit (più brevemente SDK), che supporta anche la creazione delle applicazioni per iPad e la transizione dei progetti per iPhone, quindi dei codici sorgente, verso iPad con la possibilità di ricompilarli.
La versione di SDK che consente di creare applicazioni per i Pad è la 3.2.
L'xCode non presenta grandi cambiamenti, se non per il fatto che esiste una nuova tipologia di applicazioni per adattare i progetti all'iPad, ovvero Split View-Based Application.
Anche Interface Building (il programma integrato per creare l'interfaccia grafica) si è adeguato al nuovo dispositivo e consente di creare interfacce ex-novo per iPad, oppure di migrare le interfacce già create per iPhone su iPad con il comando Interface Building, File>Transition To>Transition to iPad.
Stesso discorso per il simulatore che ora consente di simulare, avendo cura di selezionare il sistema operativo 3.2, anche su iPad oltre che iPhone.
I vecchi progetti già sviluppati su iPhone possono essere importati su iPad e fatti girare in versione nativa, quindi con l'ampiezza di display dell'iPhone, oppure adattati a tutto schermo. In questo caso, però, se l'impostazione grafica prevedeva un supporto di dimensioni inferiori (il display dell'iPhone appunto) alcune immagini potrebbero essere visualizzate in modo non ottimale.
Per il resto tutto quello che si trovava nella versione SDK 3.1.2, è presente anche nella 3.2. I vecchi progetti già compilati per iPhone vengono tranquillamente supportati e gli strumenti sono, in gran parte, già ben noti al pubblico di sviluppatori.

Attenzione, per accedere alla versione 3.2 dell'SDK occorre essere iscritti al programma di sviluppo che costa 99 dollari l'anno.

etica e pensiero LIBERO

FREE SOFTWARE

Software libero, utopie e contraddizioni. Tutto questo racchiuso nel pensiero di Richard Stallman, "cantore" del software libero e delle sue molteplici sfaccettature.

Qualcuno si potrebbe chiedere cosa ci fa Richard Stallman in una rivista dedicata alla sicurezza e alla programmazione. Ci sta. Lo posso dire con la stessa fermezza con cui affermo che il software libero, badate bene libero e non open source come ci tiene a puntualizzare lo stesso Stallman, presenta non pochi punti di congruenza con la cultura hacker.
C'è la voglia di conoscere, diffondere e rendere accessibile a tutti il software che è ormai alla base del nostro lavoro, della nostra organizzazione sociale e, perché no, della nostra vita, senza restrizioni o politiche di distribuzione che tendono a privilegiare pochi a discapito di molti.
Stallman è un'icona. Un istrione che gira il mondo per diffondere la propria filosofia. E' quanto di più simile si possa accostare ad un profeta. Un profeta della cultura informatica ma anche del libero pensiero.
Ho incontrato Stallman diverse volte, durante le sue conferenze che ha tenuto e tiene ad intervalli regolari anche nel nostro paese.
Ho scelto di proporvi una delle ultime a cui ho partecipato. Rappresenta in modo incisivo lo Stallman pensiero ed è un contributo importante alla libertà intesa in senso lato.

LA CONFERENZA

E' un Febbraio di qualche anno fa, ma potrebbe essere oggi. Il "verbo" di Stallman non è cambiato, lui professa la sua filosofia con la stessa determinazione (cocciutaggine?) di sempre. Le sue idee sono permeate da una coerenza che trova pochi riscontri.
Ma torniamo alla conferenza. Richard Stallman ha lo stesso look di altre sue appraizioni, stesso sguardo attento e curioso, stessa padronanza del pulpito, segno che il suo "show" è ormai piuttosto rodato e va in scena, replica dopo replica, in modo pressoché perfetto. La curiosità si sposta quindi di dovere sui contenuti.
Ovvia la domanda: come può il concetto di free software coniugarsi, in generale, con quello di etica e trovare campi di applicazione diversi rispetto a quello puro e semplice del codice binario proprio dell'informatica?
Ci pensa Stallman a chiarirci le idee.
"Il concetto di libertà e di collaborazione rappresentano un momento irrinunciabile nella vita di ciascuno di noi. La libertà rappresenta un bene essenziale e sulla libertà si fonda l'esperienza collaborativa che rappresenta il perno del software libero ma non solo…
Se la libertà è un valore assoluto non può che trovare riscontri applicativi anche in altri settori, questo ci lega ad un discorso di etica della libertà".

FREE SOFTWARE

"Vorrei tuttavia ricondurre la vostra attenzione sul concetto cardine del software libero. Perché il software deve essere libero? Perché non accettare le logiche spesso più comode del software proprietario?
Il discorso è semplice, perché se si insegue un'etica fondata sul valore della libertà, l'unico modo per essere corerenti con tale pensiero è quello di rivolgersi al software libero: è l'unica strada percorribile. Molti usano software proprietario, ne fanno copie anche se ciò non è legale, si tratta di un atteggiamento poco etico e non consentito. Nessuno è libero di fare e distribuire copie di un programma proprietario a meno che il programma non sia Windows XP e quella persona sia Bill Gates. (risata in sala).
Certo un vostro amico potrebbe supplicarvi di fargli una copia, è perfettamente comprensibile, si tratta di un amico in fondo, ma se utilizzate software proprietario l'unico modo che avete per non violare nessuna regola etica o giuridica e proprio quello di non avere amici.
Se ci pensate questo concetto paradossale è alla base della distinzione tra software proprietario e software libero.
Il free software favorisce rapporti collaborativi: io sviluppo un programma, lo distribuisco ai miei amici, loro apportano, se lo desiderano, delle modifiche, e lo ridistribuiscono sempre in modo libero, senza applicare restrizioni. Questa babele di collaborazioni e di interazioni rappresenta un substrato importante su cui il software cresce.
Migliora nel tempo la qualità del software medesimo e vengono rimossi via via tutti i bug, ovvero i malfunzionamenti, per merito di una comunità attenta che vive da protagonista le fasi di sviluppo.
Per questo si può tranquillamente affermare che la scelta del software libero è una scelta sicura: il software libero fa esattamente quello che noi vogliamo che faccia, è trasparente, non ci sono elementi introdotti da multinazionali per spiare i nostri comportamenti

Il discorso torna, come sempre, sulle caratteristiche "malevole" dei software.
Stallman sta preparando il solito attacco teatrale al"nemico" di sempre, lo fa versandosi una tazza di the, e mentre fissa con apparente noncuranza il piccolo filtro di tela immerso nell'acqua bollente si lascia sfuggire divertito una sigla nota a tutti.

"Windows XP (pausa studiata a tavolino e risata collettiva del pubblico), ad esempio, rappresenta uno strumento perfetto per spiare le abitudini dell'utente, attraverso di esso vengono istallati sul PC una quantità di applicativi che ci spiano, monitorano le nostre attività, fanno dei report dettagliati a Microsoft che usa questi report per capire i nostri gusti, le nostre preferenze, per venderci altro software proprietario…
Dei nomi? Windows Media Player, ad esempio, rappresenta uno strumento perfetto per monitorare le nostre preferenze in termini di visione di film e video.
Ma gli esempi si sprecano.
Inoltre, per poterci spiare questi programmi installano spyware che aprono delle backdoor nei nostri PC, ovvero delle porta aperte in cui possono pericolosamente infiltrarsi pirati telematici. Insomma mettono a grave repentaglio la sicurezza del PC.
Un altro problema dei programmi commerciali è la presenza pressoché costante di bug che vengono risolti in modo lento e graduale. Il programma esce con dei difetti che vengono faticosamente corretti dagli sviluppatori impiegando molto tempo e così esce una nuova release con nuovi problemi: un circolo vizioso.
Ricorrere al software libero significa anche affrancarsi da questa schiavitù. Se l'utente scopre un problema, può rivolgersi alla comunità di utenti che usano e sviluppano quel software per cercare di risolvere il problema, il tutto in modo collettivo e più veloce di quanto non avvenga attraverso i report ai singoli programmatori che impiegano due mesi a sanare un bug e poi vi dicono 'Va bene ora qual è il prossimo'. L'attività di test collaborativa risulta senz'altro più efficace."

Piccola nota a margine sarebbe interessante chiedere a Stallman se l'esperienza collaborativa ha una natura gerarchica o allargata e pluralista. Ma non c'è tempo per farlo anche se i più attenti tra voi avranno letto tra le righe il richiamo più che evidente alle teorie di Raymond ne "La Cattedrale e il Bazaar".
Ma torniamo al buon Stallman,

LE LIBERTÀ FONDAMENTALI

"Quindi abbracciare il software libero significa non essere più prigionieri di una tecnologia, non dipendere in modo diretto dalle decisioni e dallo sviluppo di un ristretto cerchio di persone."
"Per esser libero un software deve rispettare le libertà fondamentali:
l'utente ha la libertà di eseguire il programma per qualsiasi scopo;

Richard Matthew Stallman è fondatore della Free Software Foundation, ideatore del progetto GNU e programmatore di primo piano.

l'utente ha la libertà di modificare il programma secondo i propri bisogni: perché questa libertà abbia qualche effetto in pratica, è necessario avere accesso al codice sorgente del programma, poiché apportare modifiche ad un programma senza disporre del codice sorgente è estremamente difficile;
l'utente ha la libertà di distribuire copie del programma, gratuitamente o dietro compenso;
l'utente ha la libertà di distribuire versioni modificate del programma, così che la comunità possa fruire dei miglioramenti apportati.
Quindi io posso modificare un programma con licenza GNU e ridistribuirlo, ma non applicandovi una licenza che sia più restrittiva o, peggio, di natura commerciale.
Tutto questo trova una perfetta coerenza di intenti proprio nel fondamento di GNU."
Ma il discorso delle licenze è piuttosto complesso come suggerisce lo stesso Stallman.
"Attenzione non tutte le licenze sono rispettose delle libertà fondamentali che sono alla base del software libero. Lo è la FDL coniata proprio dalla free software foundation, ovvero la Free Documentation License, oppure la GPL, la licenza d'uso libero più diffusa che è alla base della stragrande maggioranza dei programmi basati su GNU.
Entrambe sono persistenti, nel senso che non possono essere modificate e garantiscono il "permesso d'autore" o copyleft, una riscrittura del copyright che invece di limitare le libertà dell'utente nell'uso dell'opera concede ad esso una maggiore libertà d'uso."

UNA "FERITA" APERTA

Giuro, per davvero, che non so cosa pensi realmente Stallman di Linus Torvalds. Di fatto tra i due c'è un rapporto di grande stima, lo stesso Stallman elogia il lavoro svolto da Torvalds ma ci tiene a precisare alcune "cosette" e comunque, quale che sia il punto di partenza del suo lungo discorso, uno dei punti di arrivo è sempre quello: Linux.

"Linux all'inizio, si parla del 1991, non era libero lo è diventato nel 1992 adottando la licenza GPL, in precedenza il codice veniva liberamente distribuito, ma era esplicitamente vietato ottenere un qualsiasi ritorno economico dalla sua diffusione, incluso un eventuale rimborso. Ma quando si parla di Linux sarebbe più corretto parlare, in termini generali, di sistema GNU/Linux. GNU è alla base di Linux. In realtà Torvalds ci ha in un certo senso bruciati sul tempo ideando un sistema operativo funzionate basato su librerie GNU e da altri elementi già pronti che dovevano solo confluire in un sistema operativo vero e proprio. C'era solo da portare a compimento un lavoro partito molto da lontano, ma i tempi di sviluppo si sono allungati per mille ragioni e Torvalds è arrivato prima. Non critico il lavoro di Torvalds, anzi, Linux è un ottimo prodotto, vorrei tuttavia che fosse riconosciuto in modo più marcato l'apporto dell'esperienza legata a GNU che, per chi non lo sapesse, è un acronimo ricorsivo che sta per "Gnu is not Unix".

FINALE PIROTECNICO

Dopo aver parlato di Linux, Stallman interrompe il suo discorso. Afferra una busta di plastica e con tono incerto, rivolgendosi più a se stesso che al pubblico, dice: "Bene ora vi mostrerò un altro aspetto della mia personalità…"
Ci siamo, penso, ora si traveste da Zorro o, magari, da Superman con la faccia di Bill Gates sormontata da una segnale di divieto di accesso stampigliata sul petto.
Nulla di tutto questo. Colpa mia che mi sono perso gran parte degli show di Stallman in Italia e non ho rammentato le numerose foto in internet che lo raffigurano come… il paladino della GPL.
Stallman sorridendo indossa quella che ora appare come una tunica nera e poi un curioso copricapo a forma di 33 giri colore oro che simboleggia una specie di aureola.
Il paladino della free software foundation è pronto… e si mette in posa per i fotografi.
A dimostrazione, se mai ce ne fosse stato bisogno, che le conferenze di Stallman sono sempre più uno spettacolo ben rodato e accattivante.
Si chiude con il monito di sempre, che tuttavia non manca di appassionare e di infiammare i cuori.
"Per appoggiare la causa del software libero non basta usare Linux, bisogna battersi perché la libertà, anche quella del software, sia rispettata sempre, giorno dopo giorno. Deve essere una battaglia collettiva per affermare un principio fondamentale che non deve mai essere dato per scontato.
La libertà, in fondo, è l'unica cosa di cui abbiamo veramente bisogno…"
Stretta al cuore, voglia di emulare la carica dei cavalieri di Rohan ne "Il Signore degli Anelli" contro le orde malefiche di "Gates" e la sua schiera di sistemi operativi nemici del genere umano. Ma passa quasi subito…
Vale la pena di rimarcare come anche questa volta Stallman abbia sottolineato, a proposito di libertà, il paradosso americano dove è proibito usare software liberi per vedere DVD ma solo software proprietari che supportano il DRM (Digital Rights Management), un sistema di protezione, o, come afferma Stallam a proposito della privacy violata dell'utente: Digital "Recognition" Management. Quindi se uno vuole rimanere fedele all'etica del software libero e utilizza un sistema GNU/Linux ha un unico modo per farlo: non vedere i DVD, cosa che peraltro Stallman ha ammesso di fare.
Si tratta di una visione un po' integralista: se si usa un sistema con licenza libera tipo GPL, si devono usare programmi liberi con licenza GPL. Una prova di fede davvero…

CONCLUDENDO...

E' tempo di qualche domanda. Il momento giusto per sottolineare una certa suscettibilità di Stallman che incenerisce con lo sguardo un presente che nella domanda confonde Open Source con Free Software e subito dopo rimarca: "io parlo di Free software…" . A chiudere un messaggio di speranza trasversale a chi gli chiede se il fondamento etico del free software può trovare un'applicazione in altri campi della produzione industriale diversi da quello del software. La riposta naturalmente non può che essere affermativa, un perfetto prologo ad un incontro nato in nome dell'etica universale e concluso nello stesso segno.

EMACS La storia del free software (software libero e non gratuito) inizia nel 1985, quando negli Stati Uniti Richard M. Stallman pubblica il manifesto Gnu, al termine di un lungo percorso di riflessione iniziato negli anni '70. In quell'epoca Stallman, l'ultimo custode dell'etica hacker sviluppata al MIT (Massachusetts Institute of Technology), storico centro universitario americano, incubatrice di molti importanti progetti legati a Internet, utilizza i principi assimilati nei suoi anni di permanenza all'Ia Lab, il laboratorio di intelligenza artificiale, come linee guida per la sua opera più conosciuta, un programma di editing chiamato Emacs che permette agli utenti di personalizzarlo senza limite: la sua architettura aperta incoraggia le persone ad aggiungervi nuove funzioni e a migliorarlo senza sosta. Stallman distribuisce gratis il programma a chiunque accetti la sua unica condizione: rendere disponibili tutte le estensioni apportate, in modo da collaborare al miglioramento di Emacs, che diviene quasi subito l'editor di testi standard nei dipartimenti universitari di informatica.

Massimiliano Brasile
redazione@hackerjournal.it

exFAT
Il file system "sconosciuto"

SYSTEM

Non molto noto, né pubblicizzato, exFAT è un file system che presenta numerosi vantaggi rispetto a FAT 32 e NTFS.

Nell'agosto del 2008 Microsoft ha rilasciato un nuovo file system denominato exFAT che è stato successivamente scelto dalla SD Association come formattazione predefinita per le nuove memorie SDXC. L'idea della casa di Redmond era quella di sostituire l'ormai obsoleto FAT32 (che è tuttora scelto, ad esempio, come formattazione standard dei nuovi hard-disk) e poter gestire in maniera più flessibile le memorie a stato solido utilizzate in particolare nei sistemi embedded. Stranamente exFAT non ha suscitato molto clamore, probabilmente perché non è noto che presenti molti vantaggi sia rispetto a FAT32 che a NTFS e che sia utilizzabile oltre che con i sistemi embedded anche con desktop e notebook.

CARATTERISTICHE

exFAT somiglia molto alla FAT32, tanto da essere soprannominato FAT64, ma ha alcune differenze fondamentali:
- la struttura del Master Boot Record (MBR) non è compatibile con i precedenti FAT12/FAT16/FAT32;
- c'è soltanto una copia della FAT (precedentemente erano due) viene creata una mappa (cluster bitmap) per un'allocazione più rapida dei file;
- è presente un flag di contiguità (frammentazione) per ogni singolo file, per rendere l'accesso più rapido;
- è stata migliorata l'archiviazione contigua dei file, utile soprattutto con applicazioni multimediali come le registrazioni video;
- è stato introdotto il supporto al formato UTC (universal time code) già presente nei sistemi Unix;
- è cambiato il formato degli elementi della singola directory, con l'introduzione di nuove tabelle di meta-dati; inoltre sia la FAT che i cluster bitmap sono dei file.

Grazie a questi elementi, i vantaggi principali riguardano la possibilità di gestire file più grandi di 4GB (limite imposto per la FAT32) e si parla di un limite teorico di 64 zebibyte (ossia 64 x 2^70 byte) per la partizione; inoltre le prestazioni aumentano perché si possono gestire un maggior numero di cluster e una ricerca più rapida di cluster liberi durante le operazioni di scrittura. Non è poi da trascurare il fatto che si tratta di un file system comunque abbastanza semplice come struttura che potrebbe essere implementato anche per il DOS.

Le memorie SDXC trovano applicazione in tutti i sistemi embedded più diffusi, inclusi cellulari, mp3 player, fotocamere, personal video recorder.

Possono essere archiviati fino a 1000 file per ogni directory e viene supportato il Transaction-Safe Extended FAT (TFAT), un sistema di controllo di integrità dei dati sviluppato appositamente per memorie non rimovibili flash di tipo NAND e NOR e introdotto sempre con Windows CE 6.0.

SISTEMI CHE SUPPORTANO EXFAT

Ufficialmente Microsoft ha presentato exFAT con il lancio del primo service pack per Vista, ma in realtà lo aveva già introdotto con **WinCE 6.0** che lo supporta nativamente.
Su **Vista** occorre quindi aver installato questa mega-patch per averlo, ma solo con il service pack 2 si aggiunge anche il supporto per UTC. Permane la mancanza della gestione di controllo degli accessi (ACL).
Al contrario, **Windows 7** lo supporta nativamente (così come **Windows Server 2008**) e supera le limitazioni che ha comunque Vista nella gestione del ReadyBoost (una funzionalità che permette di usare un device formattato con exFAT come una cache ad alte prestazioni, cumulando il suo spazio con la memoria di sistema).
Su **XP** è possibile averlo a patto di avere installato almeno il service pack 2 e aggiungere una patch (vedi support.microsoft.com/kb/955704) con la quale il supporto diventa nativo al pari degli altri.

REVERSE ENGINEERING

Microsoft si è guardata bene dal rilasciare le specifiche del suo nuovo filesystem, che licenzia a caro prezzo a terzi (si parla di 300mila dollari per contratto). Dato l'interesse del mercato per Linux, ha stretto però un accordo con l'azienda open source Tuxera per sviluppare driver a pagamento per il pinguino, ovviamente closed source.
In casi simili a questo (basti pensare ai player DVD) queste scelte commerciali sono controproducenti e stimolano piuttosto che inibire la volontà di molte persone che non vogliono sottostarvi.
Così, non accontentandosi delle sole informazioni e supporti ufficiali, alcuni hacker cinesi hanno effettuato delle analisi di basso livello, scoprendo alcune cose interessanti e i mattoni di base del funzionamento del filesystem. Ad esempio, che la dimensione della root è dinamica, mentre nei precedenti filesystem era statica; così come è scalabile la struttura delle directory. Hanno dissezionato il filesystem con tecniche di analisi forense stabilendo con una certa sicurezza sia l'organizzazione dei dati che gli algoritmi di base.
Nel BOX #1 si può vedere come si pensa sia la nuova organizzazione del settore zero che evidenzia l'indirizzamento a 64-bit (QWORD all'offset 48h).

BOX #1 Struttura del settore zero di un filesystem exFAT

```
+00 H: BYTE [3] jump instruction
+03 H: BYTE [8] OEM name ( "EXFAT")
+0 Bh: BPB32 (00h)
+40 H: DWORD??? (78h)
+44 H: DWORD??? (00h)
+48 H: QWORD number of sectors of the volu-
me
+50 H: DWORD FAT sector number start
+54 H: DWORD FAT number of sectors
+58 H: DWORD start the cluster sector num-
ber
+5 Ch: DWORD volume of the number of clu-
sters
+60 H: DWORD start cluster number root di-
rectory
+64 H: DWORD Volume ID
+68 H: DWORD??? (0100h)
+6 Ch: BYTE SectorSizeShift
+6 Dh: BYTE ClusterShift
+6 Eh: BYTE??? (01h)
+6 Fh: BYTE??? (80h)
+1 Feh: WORD Signature (AA55h)
```

Con l'algoritmo alla base del checksum viene verificata l'integrità delle directory (vedi STRAPPO #1) e l'hash di un file (vedi STRAPPO #2).

STRAPPO #1:

```
WORD CalcChecksum (LPCBYTE entry)
(
     WORD chk = 0;
     int len, i;
```

```
      len = 32 * (entry [1] + 1);
      for (i = 0; i <len; i + +) (
          if (i == 2 | | i == 3)
              continue;
          chk = (WORD) (((chk <<15) | ¬
(chk>> 1)) + entry );
      )
return chk;
)
```

STRAPPO #2:

```
WORD CalcFileNameHash (LPCWSTR filename)
(
      WORD chk = 0;
      int len, i;

      len = lstrlenW (filename);
      for (i = 0; i <len; i + +) (
          WCHAR c = (WCHAR) CharUpperW ¬
((LPWSTR) filename );
          chk = (WORD) (((chk <<15) | ¬
(chk>> 1)) + LOBYTE (c));
          chk = (WORD) (((chk <<15) | ¬
(chk>> 1)) + HIBYTE (c));
      )
      return chk;
)
```

Partendo da questi hack iniziali pochi giorni dopo il rilascio di exFAT gli sviluppatori del kernel si erano già messi al lavoro per supportare su linux il nuovo filesystem. Grazie al contributo dello sviluppatore Ogawa Hirofumi, è stato fornito inizialmente un accesso in sola lettura (come venne fatto con NTFS) che non ha purtroppo avuto successivi aggiornamenti. Da agosto 2009 è partito invece un nuovo progetto di Andrew Nayenko raggiungibile all'indirizzo code.google.com/p/exfat, che è classificato attualmente come beta e viene sviluppato con una certa vitalità. L'obiettivo è quello di sviluppare un modulo stabile di tipo FUSE supportato da tutte le recenti distribuzioni linux, oltre che da **Mac OS X**. Certo è che con questo modulo si può avere accesso al filesystem exFAT da linux.

GIUDIZIO

Le funzionalità di exFAT non sono paragonabili a quelle di un filesystem con journaling come NTFS e permangono alcune limitazioni sia nei sistemi Microsoft, che le supportano in modi diversi, che chiaramente in quelli open source. Tuttavia, dato il basso carico di lavoro richiesto al processore per gestire un filesystem formattato exFAT (al contrario di quanto è richiesto da NTFS) è possibile avere aumenti di prestazioni su sia su periferiche embedded, che su sistemi non nuovissimi che girino ancora con XP (e ce ne sono molti). Il vantaggio principale rispetto a FAT32 resta la possibilità di gestire periferiche molto grandi come hard-disk di qualche Terabyte, rimanendo con una struttura abbastanza agile e più prestante sia in fase di scrittura che in fase di lettura. Non si parla ovviamente di un sistema affidabile (è pur sempre un sistema FAT), ma nel caso in cui più che della ridondanza ci interessa aumentare le prestazioni di accesso o gestire agilmente dimensioni elevate, exFAT può essere una scelta valida. Se invece ci interessa avere maggior sicurezza sui dati, per i sistemi Windows la scelta obbligata resta NTFS.

Il demone e il server

SERVER

Very Secure FTP Daemon è un server FTP tra i più sicuri in circolazione, scopriamone i pregi.

VsFTPd è un server ftp scritto con un occhio di riguardo per la sicurezza, rispetto ai suoi fratelli maggiori risulta più snello, semplice da configurare e con molte opzioni utili. Tutti fattori che hanno portato questo demone su server importanti come kernel.org (sostituendo il piu' famoso ProFTPd), kde.org, gnu.org, redhat.com etc... In particolar modo l'aspetto della sicurezza è stato raggiunto grazie ad un'accurata fase di design, all'uso dei chroot ove possibile, alle capabilities e al ridottissimo utilizzo dei privilegi di root, con routine particolarmente compatte per ridurre al minimo la possibilità di bug.

IL PROGRAMMA

Per prima cosa procuriamoci una copia del programma scaricandola da ftp://vsftpd.beasts.org/users/cevans/ (o dalla sezione download della rivista) quindi unzippiamola e compiliamola. Ho utilizzato la release 2.0.1, ma nel momento in cui scrivo è disponibile anche la 2.2.2.

```
# tar xzf vsftpd-2.0.1.tar.gz
# cd vsftpd-2.0.1
# make
```

Se tutto è andato bene, dopo qualche secondo avremo un file chiamato vsftpd di circa 90kb:

```
# ls -l vsftpd ¬
-rwxr-xr-x  1 root root 90656 Aug 30 16:40 ¬
vsftpd
```

Il server ha bisogno dell'utente nobody (tutte le operazioni non privilegiate le farà con questo user), di default dovrebbe già essere presente su tutte le distribuzioni, controlliamo che ci sia o aggiungiamolo in caso contrario:

```
# grep nobody /etc/passwd ¬
nobody:x:65534:65534:nobody:/:/bin/false
```

Nel caso non fosse presente basterebbe fare:

```
# useradd -s /bin/false nobody
```

Vsftpd ha anche bisogno della directory /usr/share/empty, quindi creiamola:

```
# mkdir /usr/share/empty
```

Questa parte è molto importante, se pensate di dare accesso anonimo al vostro server, allora dovrete creare l'utente ftp, altrimenti potete saltare questa fase:

```
# useradd -s /bin/false -d /home/ftp ftp
```

Con questo comando creiamo un utente che ha come login "ftp", la cui home si trova in /home/ftp al quale però viene preclusa la facoltà di loggarsi sul sistema (ad esempio tramite ssh o telnet) questo perché come shell ha /bin/false. E' importante che la home directory dell'utente ftp non sia scrivibile dall'utente stesso, e risolviamo semplicemente cambiando l'owner e il gruppo della directory:

```
# chown root:root /home/ftp
# chmod 755 /home/ftp
```

La sua home appartiene ora al root, ed è scrivibile solo dal root stesso.
Fatto ciò siamo pronti ad installare il nostro server ftp sul sistema, perciò basta digitare:

```
# make install
```

Vsftpd è ora pronto a partire (lo troverete in /usr/sbin/vsftpd o /usr/local/sbin/vsftpd).
Prima di configurarlo dobbiamo scegliere se far girare il nuovo server tramite inetd/xinetd oppure farlo girare in standalone mode.
La differenza risiede nel fatto che utilizzandolo in standalone mode, dovremo avviarlo quando la nostra macchina fa il boot, poi il programma resterà in ascolto per le richieste ftp. Utilizzando inetd/xinet il server verrà avviato soltanto quando qualcuno fa esplicita richiesta di connessione alla porta ftp. A voi la scelta che più risulta comoda.
Se avete scelto di usarlo in standalone mode allora potete saltare la prossima lezione, altrimenti...

INETD E XINETD

Se utilizzate inetd non dovrete far altro che aprire /etc/inetd.conf ed aggiungere:

ftp stream tcp nowait root /usr/sbin/tcpd /usr/sbin/vsftpd /etc/vsftpd.conf

al posto della riga che avviava il vostro vecchio server ftp (o aggiungetela semplicemente se non avevate un server ftp). In questo modo inetd saprà che alla richiesta di una connessione FTP dovrà aprire vsftpd.
Se invece utilizzate xinetd potete semplicemente creare un file chiamato vsftpd nella directory dove risiedono i file di configurazione dei servizi di xinetd (in genere /etc/xinetd.d/) ed inserirci:

```
service ftp
{
        socket_type     = stream
        wait            = no
        user            = root
        server          = /usr/sbin/vsftpd
        server_args     = /etc/vsftpd.conf
        log_on_success  += DURATION USERID
        log_on_failure  += USERID
        nice            = 10
        disable         = no
}
```

La configurazione è simile a quella dell'inetd tranne per il fatto che specifichiamo con "nice" la priorità con cui far girare il server (-20 è la priorità più alta, 19 è la più bassa) e in più chiediamo a xinetd di loggare ogni richiesta di connessione al server.
Se in futuro voleste disabilitare il server (perché ad esempio lo state aggiornando) settate "disable" a "yes" e poi reimpostatelo a "no" non appena sarete pronti a farlo tornare online.
In entrambi i casi ricordate di inserire il path corretto in cui si trova il demone, e soprattutto: ravviate inetd/xinetd con questo comando:

```
# killall -HUP `pidof inetd`
```

Oppure:

```
# killall -HUP `pidof xinetd`
```

Diciamo così a inetd/xinetd di rileggere il file di configurazione per tener conto degli ultimi cambiamenti nella configurazione.

CONFIGURARE IL SERVER

Passiamo ora al server ftp, il file di configurazione può chiamarsi come vogliamo, e lo possiamo mettere dove più ci è comodo, partiamo con un file vuoto di nome vsftpd.conf in /etc:

```
# cd /etc
```

```
# touch vsftpd.conf
```

Editiamo il file appena creato aggiungendo le seguenti opzioni, una per riga:

```
background=YES
```

in questo modo il server si metterà automaticamente in modalità demone e ritornerà il controllo alla shell.

```
listen=YES
```

Listen possiamo settarlo a NO se utilizziamo inetd/xinetd, altrimenti lo setteremo a YES, così, una volta avviato, si metterà da solo in ascolto sulla porta ftp.

```
anonymous_enable=YES
```

Questa opzione è importante, mettendola a YES consentiamo l'accesso anonimo all'ftp, mettendola a NO, ovviamente, la disabilitiamo. Se il vostro sarà un ftp pubblico potrete tenerla a YES, altrimenti vi consiglio caldamente di disabilitarla.

```
local_enable=YES
```

local_enable indica al server se consentire o meno l'accesso agli utenti locali, tenendolo a YES tutti gli utenti con una shell valida, presenti in /etc/passwd, saranno in grado di fare il login sull'ftp, in caso contrario non gli sarà consentito di usare il server.

```
write_enable=YES
```

wrte_enable impostato a YES abilita i seguenti comandi: STOR, DELE, RNFR, RNTO, MKD, RMD, APPE, SITE. Se pensate di fare un server ftp in sola lettura (dal quale gli utenti possono solo scaricare) allora settatelo a NO, altrimenti potete metterlo a YES.

```
local_umask=022
```

la umask sono i permessi di default con cui vengono creati i file nella directory del server, quella di default di vsftpd è un po' restrittiva (tutti i file vengono creati con i permessi 700) settandola a 022 i file vengono creati con i permessi 755, cioè scrivibili e leggibili dal proprietario, leggibili dal gruppo e da tutti gli altri.

```
anon_upload_enable=NO
```

settando a NO questa opzione (valida solo se abbiamo abilitato il login anonimo), non consentiamo agli utenti anonimi di mandare file sul nostro server, se desiderate settarla a YES ricordate di creare all'interno di /home/ftp una directory (ad esempio /home/ftp/incoming) scrivibile dall'user ftp.

```
anon_mkdir_write_enable=NO
```

evitiamo che gli utenti anonimi creino delle directory, se avete particolare fiducia potete anche settarla a YES.

```
dirmessage_enable=YES
```

dirmessage_enable è un'opzione puramente estetica, abilitandola gli utenti riceveranno un messaggio se nella directory in cui stanno accedendo è presente il file .message con dentro scritto qualcosa (potete ad esempio inserire un .message nella directory incoming, con scritto qualcosa tipo: "In questa directory sono consentiti solo gli upload", oppure potete inserire una breve descrizione delle directory se avete intenzione di fare un server contenente molte cartelle). Settatela a NO se non prevedete di usarla.

```
connect_from_port_20=YES
```

questa opzione fa sì che il flusso di dati per il server abbia luogo dalla porta 20 (che è la porta ftp-data), in realtà possiamo disabilitarla (consentendo al server di utilizzare qualche privilegio in meno) ma qualche client potrebbe lamentarsi nel vedere che i dati arrivano da una porta non-standard, credendo magari ad un tentativo di hijacking. Per motivi di compatibilità possiamo tenerla abilitata.

```
xferlog_enable=YES
```

xferlog fa sì che venga creato un log contenente tutti i file uploadati e scaricati dal server. Se prevedete un massiccio traffico sul server potete disabilitarla, anche se è una buona idea tener traccia di cosa succede sui nostri server.

```
xferlog_std_format=YES
```

in questo modo diciamo a vsftpd che vogliamo il log scritto con il formato standard xferlog, questa opzione risulta utile se utilizzate dei parser per i log che tracciano delle statistiche, se non ne fate uso e volete provare un formato nuovo settatela a NO.

```
xferlog_file=/var/log/vsftpd.log
```

questo è il path in cui verrà creato il file di log, che in questo caso abbiamo chiamato vsftpd.log.

```
idle_session_timeout=600
```

questo parametro indica dopo quanto tempo di inattività l'utente viene disconnesso, se prevedete molti utenti allora potete abbassare il limite anche a 30-60 secondi piuttosto che a 600.

```
data_connection_timeout=300
```

questi, invece, sono i secondi che il server attenderà un flusso di dati prima di annullare la connessione. Data_connection_timeout torna utile nel caso che un utente venga disconnesso dalla rete, per cause indipendenti da noi, durante l'upload o il

download di dati.

```
nopriv_user=nobody
```

dopo aver eseguito il bind sulla porta da noi scelta, il server ftp (per ovvi motivi di sicurezza) deve lasciare i privilegi dell'utente root e passarli a qualcun altro, con nopriv_user specifichiamo quale sarà l'utente dal quale verranno ereditati i privilegi, nobody è in genere l'utente con meno privilegi di tutto il sistema.

```
async_abor_enable=NO
```

questa opzione fa sì che il server riconosca le richieste di ABOR (interruzione) asincrone, possiamo settarla a NO per motivi di sicurezza, anche se i client più vecchi potrebbero risentirne.

```
ascii_upload_enable=YES
ascii_download_enable=YES
```

queste due opzioni sono piuttosto delicate, indicano al server se consentire o meno gli upload o i download in modalità ASCII, potete settarle a NO se prevedete di non far mai trasferimenti di file ascii (vale a dire: file di testo). Settandole a YES risolvete il problema dei file ascii, con la consapevolezza che se un utente iniziasse a chiedere ripetutamente il SIZE di un file molto grande, consumerebbe gran parte delle vostre risorse causandovi un vero e proprio DoS.

```
ftpd_banner="Welcome to ¬
xyz"
```

ftpd_banner è semplicemente il banner che apparirà in fase di connessione, settatelo come volete, evitando di scrivere il nome e la versione del vostro ftp server, in modo da rendere un tantino più difficile un eventuale attacco da parte di qualche utente, o peggio, da parte di qualche worm.

```
chroot_local_user=YES
chroot_list_enable=YES
chroot_list_file=/etc/vsftpd.
chroot_list
```

le ultime tre opzioni sono molto importanti, la prima serve a dire se vogliamo chrootare gli utenti nelle rispettive home directory (in pratica agli utenti del sistema loggati sul vostro server, non sarà concesso di uscire dalla loro home directory durante la sessione ftp), settando la prima opzione a YES diciamo al demone che tutti gli utenti devono essere chrootati. La seconda opzione specifica che abbiamo una lista di utenti che NON vogliamo chrootare (supponiamo di avere un account amministrativo su un server remoto, sarebbe giusto chrootare gli utenti, ma noi stessi dovremmo poter gironzolare liberamente per il sistema).
Il path di questa lista è specificato dalla terza opzione.
Il file ovviamente va riempito con i nomi degli utenti che vogliamo lasciar liberi, uno per riga.

Possiamo chiudere a questo punto il file di configurazione, le altre opzioni che vi mostro serviranno solo in particolari casi, ma visto che tornano comunque utili, ne farò una breve panoramica:

`max_clients=x`
il numero massimo di connessioni accettato dal server.
`max_per_ip=x`
il numero massimo di connessioni che può stabilire un singolo IP.
`Pasv_max_port=x`
`Pasv_min_port=x`
La porta più alta e più bassa che verranno utilizzate nelle connessioni passive (utilissima se il firewall vi concede solo un determinato range).
`local_max_rate=x`
La massima velocità di trasferimento consentita ad ogni utente locale.
`Ssl_enable=Y/N`
Viene abilitato il trasferimento crittografato dei dati, logicamente dovete aver OpenSSL installato, e i client devono essere predisposti per accettare connessioni SSL.
`No_anon_password=Y/N`
Se settata a YES agli utenti anonimi non verrà chiesta la password ma solo il login.
`Ls_recurse_enable=Y/N`
Abilita o disabilita "ls –R" che può consumare molte risorse se il vostro server contiene molti dati.
`Download_enable=Y/N`
Se settato a NO agli utenti non sarà consentito di scaricare nulla dal vostro server.
`Listen_ipv6=Y/N`
Se abilitato il server riceverà soltanto connessioni ipv6 e non ipv4

Menzione a parte fa fatta per un'opzione piuttosto particolare: `hide_file`.
Grazie a questa opzione possiamo infatti nascondere uno o più file che rispondono ad un determinato pattern. Hide_file utilizza una parte delle regular expression per valutare se il file e' da nascondere o meno, un esempio che ci viene fornito dalla documentazione è:

```
hide_file={*.mp3,. ¬
hidden,que*o,hide*,h?}
```

se configurassimo il nostro ftp server in queste maniera, verrebbero nascosti (ma sarebbero comunque accessibili se un utente ne conoscesse l'esatto nome) i file:

1. con estensione .mp3
2. il cui nome sia .hidden
3. che iniziano per "que" e finiscono in "o" (questo, quello, quequero)
4. che iniziano con "hide" (hidefile, hide1, hideme, hideout)
5. il cui nome sia composto da una "h" e da un carattere qualunque (h1, h2, h3, ha, hh, hQ...)

Ora che il nostro file di configurazione è pronto possiamo avviare il server ftp. Ho scelto di utilizzare la modalità standalone, quindi per avviare il server basterà fare:

```
# /usr/sbin/vsftpd /etc/ ¬
vsftpd.conf
```

Ovvero il path del server seguito dal path del file di configurazione.
Per verificare che tutto sia andato come prevedevamo digitiamo:

vsftpd

Probably the most secure and fastest FTP server for UNIX-like systems.

Main index

- About vsftpd
- Features
- Online source / docs
- Download vsftpd
- Who recommends vsftpd
- vsftpd security
- vsftpd performance

News

Kindly hosted by Mythic Beasts Ltd.

Other links you may be looking for

- My security blog: http://scarybeastsecurity.blogspot.com/
- My security advisories: http://www.scary.beasts.org/security/

Nov 2009 - vsftpd-2.2.2 released

- vsftpd-2.2.2 is released - with a fix for a regression where heavily loaded sites could see the occasional client get kicked out just after connect. This regression is believed to be introduced in v2.1.0, affecting the inbuilt listener mode. Please refer to the v2.2.2 Changelog and vsftpd FAQ (frequently asked questions) for a list of common questions!
- After numerous requests, I now have a PayPal button for donations. If you use vsftpd, like it, and think it's worthy of a donation, then click on the Paypal button on the left of the page.
- ftp.freebsd.org switched to vsftpd.
- vsftpd tarballs are now GPG signed by me.

Sept. 2003 - Is any server other than vsftpd safe?

- ProFTPd suffers serious security hole - Sep 2003
- wu-ftpd suffers serious security hole - Jul 2003.
- lukemftpd (as a random example from many), via trust of realpath(), suffers serious security hole - Aug 2003.

ftp.redhat.com is powered by vsftpd for performance reasons - see below

ftp.openbsd.org is powered by vsftpd because it needs to be very secure! - see below

SuSE Someone sent me this green lizard.. (ftp.suse.com)

PER ULTERIORI INFORMAZIONI SI PUÒ FARE RIFERIMENTO AL SITO: HTTP://VSFTPD.BEASTS.ORG.

```
# ps aux | grep vsftpd
root      6600  0.0  0.1  1868
528 ?         Ss    Aug 30 0:00
/usr/sbin/vsftpd /etc/vsftpd/
vsftpd.conf

$ ftp localhost
Connected to localhost.
220 "Welcome to xyz FTP ¬
server."
Name (localhost:xyz):
SSL not available
331 Please specify the ¬
password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer¬
files.
ftp> ls
200 PORT command successful. ¬
Consider using PASV.
```

Se avete scelto di utilizzare il server tramite inetd/xinetd vi basterà eseguire l'ultimo comando: "ftp localhost" (se siete in locale) oppure "ftp ip_del_server" (nel caso il server si trovi su una macchina remota). Inetd/xinetd avvieranno il server per voi e vi verrà chiesto username e password.
Provate anche a scaricare un file e poi aprire il log in /var/log/vsftpd.log per vedere se il sistema di logging funziona correttamente, dovreste ottenere qualcosa di simile a:

```
# tail -n 3 /var/log/ ¬
vsftpd.log
Mon Aug 30 01:07:24 2009 1
192.168.1.45 80284 /home/ftp/
ia.zip b _ o r honeypot ftp 0
* c
Mon Aug 30 01:08:10 2009 6
192.168.1.45 6911796 /home/ftp/
epr.pdf b _ o r honeypot ftp 0
* c
Mon Aug 30 01:09:42 2009 27
192.168.1.45 137728 /home/ftp/
polansky.pdf b _ o r honeypot
ftp 0 * c
```

Il log va letto in questa maniera: il primo campo è il giorno della settimana, seguito dal mese e dal giorno (in forma numerica), l'ora e l'anno, l'ip di chi ha fatto la richiesta di connessione, la dimensione in byte del file prelevato/uploadato, il path assoluto del file prelevato/uploadato, il tipo di trasferimento (a se ascii, b se binario), se sono state eseguite azioni particolari dal server (C se il file era compresso, U se il file era decompresso, T se il file era un archivio TAR, _ se non è stata eseguita nessuna azione speciale), la direzione (o se il file è andato dal server all'utente, i se è andato dall'utente al server), il metodo di login dell'utente (a se anonimo, g se guest, r se tramite username legittimo contenuto in /etc/passwd), l'username dell'utente, il servizio utilizzato (ftp in questo caso) il metodo di autenticazione utilizzato (0 se non è stato utilizzato nessun metodo di autenticazione, 1 se è stata usata l'autenticazione RFC931), l'id ritornato dall'autenticazione (* se non è tornato nessun ID, come nel nostro caso, visto che non abbiamo utilizzato alcuna autenticazione) ed infine lo stato del trasferimento (c se completo, i se incompleto).

CONCLUSIONI

Come avete visto vsftpd consente una configurazione estremamente capillare e nonostante questo resta molto intuitivo.
E' un programma molto versatile, grazie al quale diventa semplicissimo aggiungere il supporto SSL e il supporto ipv6, si nota molto l'approccio alla sicurezza introdotto dall'autore, che ci dà la possibilità di controllare anche quei comportamenti generalmente poco frequenti, ma che, comunque, potrebbero essere utilizzati per attaccare o rallentare la nostra macchina. Nonostante le dimensioni contenute, si presenta come un ottimo rivale dei suoi fratelli maggiori proftp e wuftp essendo dotato di ottima scalabilità e velocità.

di Giovanni Federico
giovanni.federico@isek.it

PARTE I

HOST-BASED INTRUSION DETECTION & LOG MONITORING OSSEC

SICUREZZA In questo articolo presentiamo un'introduzione ai processi di difesa autonomi per il controllo del rischio ed il monitoraggio dei log con OSSEC utilizzando OpenBSD ed il suo packet filter.

L'esponenziale crescita del mercato che ruota attorno all'Information & Communication Technologies ed il vistoso aumento di reti per l'interconnessione di sistemi informativi, hanno delineato, nel corso degli anni, una radicale trasformazione del nostro modo di vivere che interessa ogni singola azione quotidiana.
Alzarsi al mattino ed accompagnare il caffè leggendo la nostra casella di posta elettronica è un'operazione di prassi ormai, così come lo è gestire centinaia o migliaia di contatti attraverso i vari social network ed i moderni software di messaggeria istantanea.

Se pensiamo poi ai sistemi di commercio elettronico, alla gestione dei conti correnti con un semplice click, all'opportunità di organizzare viaggi, prenotare aerei e quant'altro, giocare in borsa, pagare il casello autostradale e veicolare eccellentemente pubblicità attraverso siti e portali, appare quantomeno scontato presumere che **il mondo dei bit sia qualcosa di molto meno astratto ed avulso dalla realtà rispetto a quello che si potrebbe pensare.**

Questo fiorente mercato ha avuto modo di creare nuove opportunità di lavoro, vere e proprie figure professionali che fino a qualche tempo fa non erano minimamente immaginabili.

Se infatti fino a 20 anni fa la prerogativa principale, in termini di risorse umane, di un'azienda di medio-alto livello era quella di avvalersi di un buon General Manager, ora risulta del tutto parificata, in termini di importanza aziendale, la figura dell'IT Manager.

In virtù di questo processo simbiotico tra uomo e macchina diviene chiaro percepire come l'interesse alle tematiche relative al mondo della sicurezza informatica avvolga un po' tutti, addetti ai lavori e non.

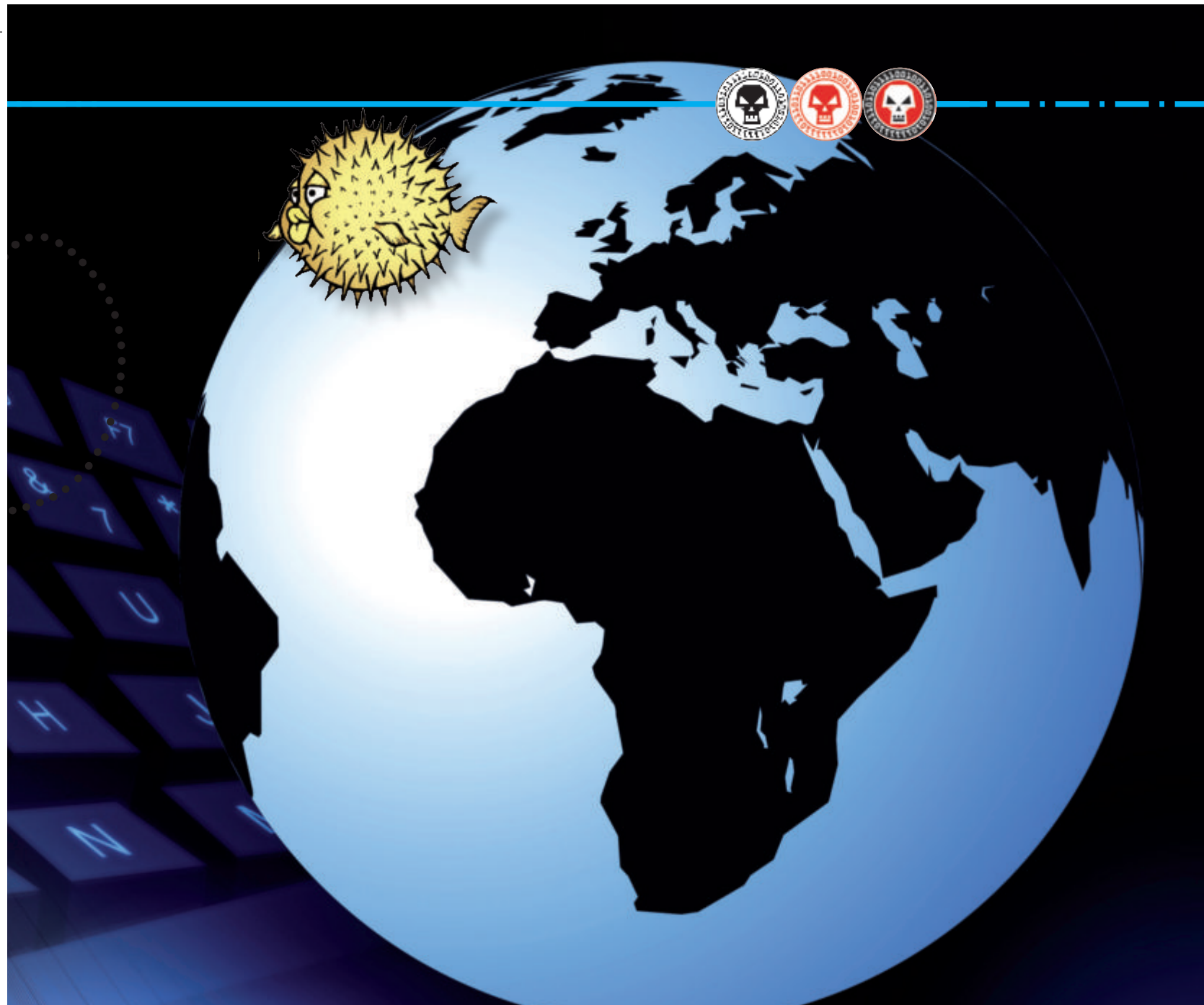
Anche sfruttando questa forte richiesta è stato creato **un modello di business da zero**: basti pensare all'industria dei produttori di software antivirus, firewall, IDS e via dicendo e alla continua nascita di aziende che fanno della sicurezza dell'informazione, della privacy e della gestione del rischio il loro core business.

Ma quali sono le metodologie generalmente adottate per monitorare costantemente lo stato di sicurezza di un'intera infrastruttura IT ? Che cosa significa rendere sicura un'infrastruttura IT ? Più in generale, cosa si intende con il termine "sicurezza" ?

In un articolo proposto su HJ 194 abbiamo introdotto i tre pilastri fondamentali dell'IT Security che riprendiamo anche in questa sede, ovvero: confidenzialità, integrità e disponibilità.

Offrire sicurezza significa quindi **governare il rischio** in base ad un'organizzazione tecnica e logistica che tenga conto di questi tre inscindibili presupposti e che utilizzi tutte le risorse disponibili (umane, software e hardware) per farlo, presupponendo che:

Gli host del nostro network debbano essere accessibili solo da parte di chi è autorizzato.
Le informazioni in essi contenuti debbano essere modificabili solo da chi autorizzato secondo schemi e modalità definite dall'Amministratore.
Debbano essere sempre accessibili da parte di chi è autorizzato.

Il processo di Auditing si suddivide in fattore proattivo (**analisi dei sistemi**) e reattivo (**sviluppo delle policy di sicurezza adeguate**): individuare i principali punti focali del rischio, analizzando le maggiori criticità della nostra infrastruttura e sviluppando un'opportuna strategia di **contenimento** costituisce un

attributo fondamentale del nostro percorso.

Avendo già analizzato nel corso dei passati numeri della rivista le modalità di attacco alle quali i nostri sistemi sono esposti e le metodologie generalmente adottate da parte di incursori per farvi breccia (fattore proattivo), focalizzeremo la nostra attenzione su uno degli aspetti più cruciali: **la definizione delle politiche atte alla gestione del rischio** (fattore reattivo), ricordandoci che la protezione offerta deve essere direttamente proporzionale alle criticità rilevate in una precedente fase di auditing del nostro network, dall'interno e dall'esterno (valutando, quindi, le risposte ai **penetration test** effettuati).

Vedremo quindi come i concetti alla base **del corretto monitoraggio dei log** possano permetterci di ricevere un immediato riscontro delle informazioni gestite dai nostri sistemi individuando i principali campanelli di allarme e come, attraverso questi, **bloccare sul nascere accessi non autorizzati e tentativi di incursione.**

Analizzeremo pertanto OSSEC, uno dei più potenti **Host-based Intrusion Detection System** (HIDS) offerti dalla comunità del software libero osservando come, attraverso l'ausilio del sistema operativo **OpenBSD** ed il suo firewall integrato (PF) sia possibile **realizzare un complesso ed autonomo meccanismo di protezione perimetrale della nostra rete e degli host che la compongono.**

SCENARIO

Considerata la totale promiscuità che caratterizza le reti di qualsiasi contesto aziendale medio/grande, ci preoccuperemo di realizzare una soluzione che trovi spazio in qualsiasi scenario lavorativo e che ben si presti ad essere interoperabile con svariate tipologie di OS ed architetture.

Nella fattispecie analizzeremo una classica situazione operativa dove vedono configurarsi: un web server Apache, un servizio FTP, accesso SSH abilitato su n host, vari client

Di seguito schematizzato il contesto operativo che analizzeremo nel corso del presente articolo e la modalità di funzionamento dell'HIDS.

Microsoft Windows e GNU/Linux, un file/print server gestito con Samba, uno o più server di posta elettronica ed un server MySQL.

Opteremo pertanto per una configurazione dell'HIDS di tipo client/server (di seguito "c/s") demandando la gestione delle regole di filtraggio e degli alert ricevuti ad una macchina montante OpenBSD 4.6.

Ci occuperemo, in una seconda fase, di rendere perfettamente sincroni gli operati di PF e di OSSEC evidenziando come sia possibile, attraverso quella che è definita modalità "Active Response", rendere le operazioni di log monitoring e blocco degli attacchi real-time perfettamente autonome.

ANALISI DEI LOG

Sviluppato da Trend Micro e disponibile con licenza GPL, al momento della scrittura di questo articolo,

l'ultima versione di OSSEC disponibile per il download è la 2.3.

L'applicativo, in versione Client, funziona su sistemi operativi Microsoft Windows, GNU/Linux, Solaris, BSD, AIX ed HP/UX. Per quanto riguarda il Server, invece, la compatibilità è assicurata solo per OS Unix-like.

Prima di procedere con l'installazione del server e degli agenti sui nostri host (cosa che demanderemo al prossimo numero della rivista) è necessario chiarire come l'applicativo suddivide il proprio lavoro in modo estremamente modulare.

Al momento dell'avvio OSSEC, infatti, definisce un set di demoni attivi sulla macchina con privilegi limitati al loro specifico uso: ognuno occuperà una specifica funzione.

OSSEC Analysisd

Questo modulo costituisce il cuore dell'applicativo occupandosi, come il nome stesso suggerisce, dell'analisi dei log e degli eventi (tra poco vedremo su che presupposti e come è basata l'indagine sui log). Nell'installazione c/s il processo è collocato esclusivamente sul Server (che si presuppone essere totalmente dedicato a questo utilizzo) lasciando le risorse per gli applicativi operanti sui nostri client immutate. Nelle installazioni standalone è invece naturalmente avviato anche sui client (con un aggravio in termini di risorse).

OSSEC Agentd

Attivo sui nostri client, si occupa di inviare le informazioni necessarie all'espletamento dell'analisi dei log al Server. Nell'installazione c/s è naturalmente collocato all'interno dell'agente (e quindi del Client).

OSSEC Remoted

Disponibile sul server nell'installazione c/s, si occupa di coordinare le comunicazioni tra gli agenti (client) ed il server interfacciandosi direttamente al modulo Analysisd per la successiva verifica ed analisi delle informazioni ricevute dagli host della rete.

OSSEC Monitord

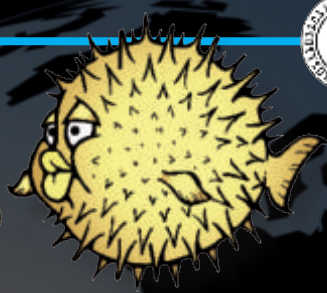
Compito del modulo è quello di controllare l'operato dei client (Agenti) ed archiviare tutti i log centralizzati prodotti in appositi archivi giornalieri.

OSSEC Log collector

Interfacciandosi con Monitord rappresenta l'effettivo worker atto al recupero dei log; per questo è naturalmente avviato con privilegi amministrativi.

OSSEC Execd e Maild

Com'è facile intuire, questi ultimi moduli si occupano dell'invio degli alert via posta elettronica (vedremo di seguito come) e dell'attivazione dei processi di difesa necessari, identificati in OSSEC sotto il nome di "Active response" (attivazione del firewall, inserimento host attaccanti in hosts.deny, scrittura delle regole di filtraggio opportune in caso di attacco direttamente nel file di configurazione del firewall).

Risulta a questo punto facile comprendere come il processo di log monitoring effettuato da OSSEC poggi le basi su un'efficiente strategia modulare all'interno della quale si colloca un'intera catena produttiva che tiene conto del recupero dei log (log collector), dell'invio di questi ultimi al server (agentd e remoted), dell'analisi degli stessi (analysisd), della segnalazione tempestiva delle criticità rilevate (maild) e dell'attivazione di meccanismi di difesa autonomi secondo schemi e modalità prefissate (execd).

Quest'ultimo punto trasforma di fatto l'applicativo da IDS ad IPS. Torneremo sull'argomento anche riprendendo alcuni concetti relativi a PF già visti nei numeri passati della rivista.

DECODERS

Tutti sappiamo che i file di log possono contenere una quantità di informazioni considerevole e che non tutte sono di nostro interesse. Avere un sistema centralizzato di log monitoring può rendere sicuramente l'analisi delle stesse molto più efficiente e permetterci di avere un quadro della situazione circostanziale mirato ed aggiornato in tempo reale. Ma come è reso possibile ciò?

Il parsing dei log è effettuato come abbiamo detto pocanzi dall'apposito modulo "Analysisd" in stretta sinergia con i dati raccolti dai vari agenti. Approfondiamo quindi il discorso vedendo come sia effettivamente effettuata l'analisi delle informazioni raccolte.

Il processo è suddiviso in tre singole fasi: **pre-decodifica**, **decodifica** ed **analisi**.

Nella prima, **le informazioni ricevute dai singoli agenti e dal server stesso sono parsate dal demone alla ricerca di "informazioni chiave"** quali orario, nome dell'applicazione, nome del sistema.

Nella seconda e terza fase viene analizzata, attraverso l'utilizzo di espressioni regolari, la restante parte del log nel dettaglio **alla ricerca di informazioni quanto più esaustive possibili riguardo al significato semantico dei riscontri rilevati** (indirizzi ip, username, errori, etc.).

Per capire meglio, esaminiamo un tipico decoder, nella fattispecie quello relativo al server web Apache, comprendendone il significato e l'impostazione:

```
<decoder name="apache-errorlog">
  <program_name>^httpd</¬
program_name>
</decoder>

<decoder name="apache-errorlog">
  <prematch>^[warn] |^[notice]¬
|^[error] </prematch>
</decoder>

<decoder name="apache-errorlog-¬
ip">
  <parent>apache-errorlog</pa¬
rent>

  <prematch offset="after_¬
parent">^[client</prematch>
  <regex offset="after_pre¬
match">^ (\d+.\d+.\d+.\d+)] </¬
regex>
  <order>srcip</order>
</decoder>
```

Il primo blocco di istruzioni identifica univocamente l'applicativo di nostro interesse da tenere "sotto controllo": in questo caso, trattandosi di Apache, saranno presi in considerazione i processi identificati dal nome "httpd".

Nel secondo blocco sono definite le operazioni di decodifica accennate prima; sono infatti analizzate le informazioni da monitorare all'interno dei log di errore del web server Apache che rispondano a tre determinate regexp (warn, notice ed error).

Nel terzo ed ultimo blocco si configura la ricerca dell'IP all'interno del medesimo file di log solo dopo l'analisi condotta dal secondo blocco e solo se vi sia presente un chiaro riferimento all'indirizzo stesso ([client ^IP^]).

Se analizziamo brevemente un tipico stralcio di log di Apache diventa molto più semplice capire il significato del decoder e la sua doppia analisi con regexp:

```
[Fri Mar  5 23:15:03 2010] [er¬
ror] [client 93.**.247.**] File¬
does not exist: ...
```

Risulta evidente a questo punto il senso delle espressioni regolari: nella prima parte (`[Fri Mar  5 23:15:03 2010]`) troviamo un chiarissimo riscontro orario (oggetto di analisi nel processo di pre-decodifica); nella seconda (`[error]`) troviamo una delle tre opzioni contemplate nel secondo blocco del decoder ed oggetto dell'analisi di decodifica relativa alla tipologia del messaggio (in questo caso di errore); nella terza (`[client 93.**.247.**]`) si completa il processo di decodifica avendo individuato anche l'IP di provenienza attraverso l'ok ricevuto dall'apposita regexp; nella quarta ed ultima parte si snoda il processo di analisi semantica (il significato del messaggio è reso chiaro direttamente dal testo che accompagna il log: `"File does not exist"`).

Come facilmente intuibile, risulta decisamente semplice creare decoder in formato XML contenenti espressioni regolari personalizzate per software specifico o semplicemente per analizzare dettagliatamente eventi e file.

LIVELLI DI ALLERTA

Fin qui abbiamo definito tutto il necessario per capire il modo di interfacciarsi ai software da parte dell'applicativo completando il discorso alla base del monitoraggio dei log. Ma OSSEC è ben altro e consente di stendere un profilo del rischio attraverso i riscontri ricevuti dall'analisi dei log definendo opportuni "livelli di allerta" crescenti in base alle problematiche riscontrate.

L'operato dell'applicativo è infatti disciplinato da un set di regole predefinito.

Il formato utilizzato per la definizione del modello di lettura e analisi dei log è anche in questo caso il comune XML e trovano spazio schemi pronti all'uso per la quasi totalità dei software Open Source (Apache, Mysql, Squid, Sendmail, Samba) e buona fetta di soluzioni commerciali come Antivirus, MS Exchange e firewall (vedi box).

BOX #1: BUILT-IN RULES PRESENTI NELL'INSTALLAZIONE DI BASE

```
apache_rules.xml
mysql_rules.xml
sonicwall_rules.xml
arpwatch_rules.xml
named_rules.xml
spamd_rules.xml
asterisk_rules.xml
netscreenfw_rules.xml
squid_rules.xml
attack_rules.xml
nginx_rules.xml
sshd_rules.xml
cisco-ios_rules.xml
ossec_rules.xml
symantec-av_rules.xml
courier_rules.xml
pam_rules.xml
symantec-ws_rules.xml
dovecot_rules.xml
php_rules.xml
syslog_rules.xml
firewall_rules.xml
pix_rules.xml
telnetd_rules.xml
ftpd_rules.xml
policy_rules.xml
hordeimp_rules.xml
postfix_rules.xml
translated
trend-osce_rules.xml
ids_rules.xml
postgresql_rules.xml
vmpop3d_rules.xml
imapd_rules.xml
proftpd_rules.xml
vmware_rules.xml
local_rules.xml
pure-ftpd_rules.xml
vpn_concentrator_rules.xml
mailscanner_rules.xml
racoon_rules.xml
vpopmail_rules.xml
mcafee_av_rules.xml
roundcube_rules.xml
vsftpd_rules.xml
ms-exchange_rules.xml
rules_config.xml
web_rules.xml
ms_dhcp_rules.xml
sendmail_rules.xml
wordpress_rules.xml
ms_ftpd_rules.xml
smbd_rules.xml
zeus_rules.xml
msauth_rules.xml
solaris_bsm_rules.xml
```

Senza perderci in inutili chiacchiere consideriamo come prima un esempio pratico (stavolta con SSH) ed una delle tante regole predefinite dal software (sshd_rules.xml).

Prima di procedere introduciamo però alcune nozioni di base che ci consentiranno di capire meglio i discorsi a seguire.

Ogni regola definita con OSSEC prevede un identificativo unico variabile da 100 a 99999.
Il livello di allerta definibile sulla singola regola varia da un minimo di 0 ad un massimo di 15 (in ordine ovviamente crescente rispetto al rischio individuato).
Qualora non volessimo essere informati circa un riscontro ad una determinata regola è in ogni caso possibile definire l'apposita opzione "noalert", indipendentemente dal livello di allerta stabilito.
Per limitare l'insorgenza di falsi positivi è possibile definire alcuni parametri specifici: è ad esempio consigliabile individuare una certa frequenza nei log prima di generare un alert attraverso l'apposita opzione "frequency" così come potrebbe essere interessante analizzare specifici archi temporali tra un errore e l'altro con la direttiva "timeframe".
Generato un alert, è inutile generarne un'altro per la stessa evenienza in un arco temporale ristretto che presupporrebbe lo stesso riscontro; a tal fine può rivelarsi utile l'adozione della direttiva "ignore" nel corpo della regola.

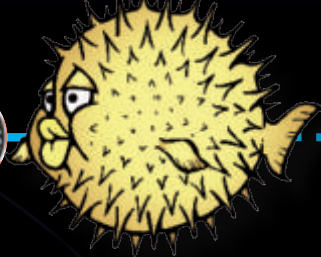
Detto questo, passiamo all'analisi di una delle tante regole per SSH, come sopra menzionato, che meglio chiarifichi i punti appena citati, occupandosi, nello specifico, di individuare i tentativi di accessi brute force (più in là vedremo anche come bloccarli sul nascere):

```
<rule id="5700" level="0"¬
noalert="1">
        <decoded_as>sshd</deco¬
ded_as>
        <description>SSHD¬
messages grouped.</description>
</rule>
```

In questo primo blocco definiamo l'id univoco da assegnare ad un semplice contenitore di messaggi relativi a SSH (e quindi identificati attraverso l'apposito decoder, come visibile alla seconda riga) definendo un livello di allerta minimo (**level="0"**) e la non notifica dello stesso (**noalert="1"**).

Questo ci servirà per realizzare alcune regole "a catena" attraverso l'apposita direttiva "**if_sid**" che vedremo di seguito. **In altre parole, realizzeremo un set di regole concatenato del tipo:**

1- Processo SSH attivo (no alert) - ID: 5700
2 - Login (no alert)
3 - Login errato (primo alert, semplicemente: "Accesso negato") - ID: 5710 < > 5700
4 - Fino al quinto login errato nell'arco di 2 minuti, stesso IP (stesso discorso) - ID: 5710 < > 5700
5 - Sesto login errato in 2 minuti, stesso IP (nuovo alert, brute force in corso) - ID: 5712 < > 5710 < > 5700

Procediamo quindi con il secondo blocco (*punto 3 del nostro set di regole*):

```
<rule id="5710" level="5">
        <if_sid>5700</if_sid>
        <match>illegal¬
user|invalid user</match>
        <description>Attempt to¬
login using a non-existent¬
user</description>
        <group>invalid_¬
login,authentication_failed,</¬
group>
</rule>
```

Qui definiamo il primo alert generato in caso di login errato. Vediamo quindi l'uso della direttiva "if_sid" accennata prima, volta a definire la relazione di dipendenza con la regola 5700 (del resto, un errore relativo all'autenticazione su SSH difficilmente potrebbe generarsi se il servizio non fosse attivo).

Attraverso la direttiva "**match**" scriviamo quindi l'apposita espressione regolare finalizzata all'identificazione della problematica in essere. Questa definisce esattamente le parole da ricercare all'interno dei log generati dal software preso in considerazione (in questo caso SSHd) affinché sia richiamata la regola stessa.

Nel tag "**description**" invece offriamo una descrizione di significato compiuto che sarà poi riportata sull'alert vero e proprio. In ultima istanza, alla voce "**group**", definiamo la tipologia entro cui la regola prende forma.

Vediamo quindi l'ultimo blocco (*punto 5 del nostro set di regole*):

```
<rule id="5712" level="10"¬
frequency="6" timeframe="120"¬
ignore="60">
        <if_matched_sid>5710</¬
if_matched_sid>
        <description>SSHD brute¬
force trying to get access to¬
the system.</description>
        <same_source_ip />
        <group>authentication_¬
failures,</group>
</rule>
```

Per questo blocco (che rappresenta sostanzialmente un'integrazione del secondo), come ci aspettavamo, la relazione di dipendenza è impostata con la regola 5710 ma stavolta è più forte, ereditando attraverso la direttiva "**if_matched_sid**" tutto il corpo-regola definito prima ed aggiungendo ulteriori specifiche.

Presenta un livello di allerta elevato (10) e viene attivata unicamente dopo 6 riscontri alla regola 5710 (**frequency="6"**) generati in una frazione di 2 minuti (**timeframe="120"**) .
Scattato l'alert, infine, il sistema non riconsidererà la regola per un minuto (**ignore="60"**).

L'utilizzo della direttiva "**same_source_ip**" determina l'attivazione del blocco solo in presenza di richieste provenienti dallo stesso IP.
I tag "description" e "group" si commentano a questo punto da soli.

L'ALERT GENERATO

In tre semplici passaggi abbiamo definito una precisa metodologia per l'individuazione dei tentativi di accesso non autorizzati via SSH ai nostri host. Vediamo quindi un tipico alert inviato dal demone "maild" alla nostra casella di posta elettronica durante un tentativo di attacco brute force:

```
OSSEC HIDS Notification.
2010 Mar 07 00:58:04

Received From: www->/var/log/
authlog
Rule: 5712 fired (level 10) ->
"SSHD brute force trying to get
access to the system."
Portion of the log(s):

Mar  7 00:58:03 www sshd[22643]:
Failed password for invalid user
marine from 123.125.127.207 port
35068 ssh2
Mar  7 00:58:03 www sshd[22643]:
Invalid user marine from
123.125.127.207
Mar  7 00:57:59 www sshd[5382]:
Failed password for invalid user
marine from 123.125.127.207 port
31461 ssh2
Mar  7 00:57:59 www sshd[5382]:
Invalid user marine from
123.125.127.207
Mar  7 00:57:54 www sshd[29559]:
Failed password for invalid user
marine from 123.125.127.207 port
28051 ssh2
Mar  7 00:57:54 www sshd[29559]:
Invalid user marine from
123.125.127.207
Mar  7 00:57:50 www sshd[19892]:
Failed password for invalid user
marine from 123.125.127.207 port
24683 ssh2
```

Come immaginavamo, nell'alert, oltre i dettagli del caso, è riportato un chiaro riferimento alla regola 5712 (evidenziato).

Viene quindi successivamente riportato lo spezzone di error log del web server incriminato, utile per identificare all'istante orari ed IP di provenienza dell'attacco. Senz'altro utile, no ?

REGOLE AD HOC

Ora che abbiamo capito come funziona l'architettura dell'applicativo e su che schemi poggia il suo funzionamento possiamo lanciarci finalmente nella parte più creativa dell'articolo: la scrittura di regole personalizzate.

In questa sede vedremo, a titolo esclusivamente esemplificativo, come sia possibile scrivere una specifica regola che ci consenta di monitorare costantemente le porte aperte su un determinato host (interno o esterno alla rete) utilizzando il comodo e semplice **nmap** ed interfacciandolo con l'HIDS.

L'applicativo infatti è in grado di leggere senza alcun problema i log in formato grepable (*da nmap: "-oG nomefile")* di nmap come qualunque altro log di sistema.

Gli alert saranno quindi generati nel formato standard OSSEC ed inviati periodicamente via posta elettronica ogni qualvolta subentrino nuove informazioni interessanti (porte aperte ma precedentemente chiuse o viceversa).

Per i più pigri, ricordiamo che tutti i file XML di seguito analizzati sono disponibili online al sito www.hackerjournal.it.

SCANSIONE HOST

Per la scansione dell'host (in questo caso interno alla rete e precisamente 192.168.1.100) opteremo per l'analisi delle porte UDP e TCP via connect() utilizzando l'interfaccia di rete interna del firewall (nel nostro caso "em0") ed esportando il risultato su un file di testo ("*/var/log/nmap.log*") che sarà poi analizzato da OSSEC:

```
nmap --append_output -sT -sU¬
-e em0 -oG /var/log/nmap.log¬
192.168.1.100
```

Testiamo in prima battuta il funzionamento di nmap, ottenendo il consueto report:

```
PORT      STATE          SERVICE
113/tcp   open           auth
...
500/udp   open|filtered isakmp
...
MAC Address: 40:61:**:**:**:1D
(Unknown)
```

Aggiungiamo quindi una riga alla crontab del sistema che si occupi di avviare nmap ogni 30 minuti:

```
$ sudo crontab -e
```

```
30 * * * * nmap --append_output¬
-sT -sU -e em0 -oG /var/log/¬
nmap.log 192.168.1.100
```

Segnaliamo ora ad OSSEC il percorso relativo al file generato specificando nel tag "**log format**" la stringa nmapg; ciò farà intendere all'HIDS che si tratta di output in formato grepable di nmap:

```
<localfile>
    <log_format>nmapg</log_¬ for-
mat>
    <location>/var/log/nmap.¬
log</location>
</localfile>
```

Dopo un necessario riavvio di OSSEC, una rapida occhiata agli alert ci conferma immediatamente la riuscita dell'opera:

```
# tail -f /var/ossec/logs/alerts/
alerts.log
...
** Alert 1268013722.7319: mail  -
ossec,hostinfo,
2010 Mar 08 03:02:02 www->/var/
log/nmap.log
Rule: 581 (level 8) -> 'Host in-
formation added.'
Src IP: (none)
User: (none)
Host: 192.168.1.100 (), open
ports: 135(tcp) ...
```

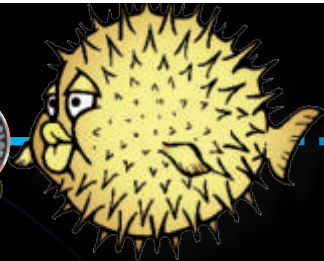
Se invece siamo restii alla shell, una mail inviata dal sistema ci notificherà immediatamente le stesse informazioni:

```
OSSEC HIDS Notification.
2010 Mar 08 03:02:02

Received From: www->/var/log/
nmap.log
Rule: 581 fired (level 8) -> "Host
information added."
Portion of the log(s):

Host: 192.168.1.100 (), open
ports: 135(tcp) ...
```

Vediamo invece cosa accade nel momento in cui, sempre sull'host considerato, **viene aperta un'ulteriore porta rispetto a quelle identificate dal primo alert.** Proviamo, banalmente, ad aprire la porta **50000** con un semplice script:

```
$ cat bind.c
#include <sys/socket.h>
#include <netinet/in.h>

int main() {
       int s, c;
       struct sockaddr_in s_¬
addr, c_addr;
       int c_len = sizeof(c_¬
addr);

       s_addr.sin_family = AF_¬
INET;
       s_addr.sin_port =¬
htons(50000);
       s_addr.sin_addr.s_addr =¬
INADDR_ANY;
       s = socket (AF_INET,¬
SOCK_STREAM, 0);

       bind (s,(struct¬
sockaddr*)&s_addr,sizeof(s_¬
addr));
       listen(s,1);
       accept(s,(struct ¬
sockaddr*) &c_addr, &c_len);
}

$ gcc bind.c -o bind
$ ./bind
```

Puntuale come un orologio svizzero, scattata la mezzora del cronjob definito prima, un apposito alert ci informerà sia da shell che via posta elettronica circa il cambiamento avvenuto rispetto alle porte inizialmente aperte sull'host:

```
# tail -f /var/ossec/logs/alerts/
alerts.log
...
** Alert 1268017859.7693: mail  -
ossec,hostinfo,
2010 Mar 08 04:10:59 www->/var/
log/nmap.log
Rule: 580 (level 8) -> 'Host in-
```

```
formation changed.'
Src IP: (none)
User: (none)
Host: 192.168.1.100 (), open
ports: 135(tcp) ... 50000 (tcp)
```

Per finire, vediamo come sono definite le regole 580 e 581 oggetto di questi alert:

```
<rule id="580" level="8">
      <category>ossec</¬
category>
      <decoded_as>hostinfo_¬
modified</decoded_as>
      <description>Host  infor¬
mation changed.</description>
      <group>hostinfo,</group>
</rule>

<rule id="581" level="8">
      <category>ossec</¬
category>
      <decoded_as>hostinfo_¬
new</decoded_as>
      <description>Host¬
information added.</description>
      <group>hostinfo,</group>
</rule>
```

Il lettore, a questo punto, dovrebbe essere autonomamente in grado di interpretare il significato di queste due semplici regole.

Risulta quasi lapalissiano dire che l'unico freno imposto nella realizzazione di regole ad hoc è la nostra fantasia; come abbiamo avuto modo di vedere, OSSEC è un applicativo di comprovate possibilità e risulta facilmente modellabile.

Il lettore avrà sicuramente percepito i margini di applicazione dell'HIDS e la totale capacità di adattamento a contesti di qualsiasi dimensione e natura.

La documentazione offerta online inoltre è notevole; possiamo trovare spunti ed approfondimenti per qualsiasi evenienza con una semplice ricerca.

CONCLUSIONI

No, non ci siamo dimenticati di illustrare i passi necessari al setup del server e dei vari agenti.
Semplicemente, in questa prima parte dell'articolo, abbiamo preferire offrire una trattazione ampia tesa soprattutto all'introduzione dell'HIDS ed al suo funzionamento nel dettaglio.

Assodato ciò, risulterà sicuramente più facile ed immediato per il lettore comprendere come operativamente costruire lo scenario definito in questa sede nel corso del prossimo numero della rivista.

Per i più impazienti, come sempre, segnaliamo il forum ed il canale IRC della rivista (irc.azzurra.org - #hackerjournal). Entrambi costituiscono il miglior posto dove richiedere maggiori informazioni e trovar risposta alle proprie domande confrontandosi con gli autori e la community.

Concludiamo quindi la prima parte della nostra trattazione offrendo alcuni opportuni riferimenti citati ed utilizzati anche durante la scrittura di quest'ultima (vedi box).

BOX #2: RIFERIMENTI I PARTE

Website relativi agli applicativi ed i sistemi operativi analizzati:

Trend Micro™ OSSEC: *www.ossec.net*
OpenBSD: *www.openbsd.org*
Nmap: *www.nmap.org*
Apache httpd: *httpd.apache.org*
OpenSSH: *www.openssh.com*

Alcune riferimenti online utili:

OSSEC Manual: *www.ossec.net/main/manual/*
Regexp: *en.wikipedia.org/wiki/Regular_expression*
Beej's guide to Network Programming: *beej.us/guide/bgnet/*
Nmap official project guide to Network Discovery and Security Scanning: *nmap.org/book/toc.html*
Extensible Markup Language (XML): *www.w3.org/XML/*
IDS and IPS placement for Network Protection (by R. Drum, ISC CISSP): *tiny.cc/GJcfn*
OpenBSD 4.6 Installation Guide: www.openbsd.org/faq/faq4.html

Ulteriori riferimenti:

OSSEC Host-Based Intrusion Detection Guide by A. Hay, D. Cid, R. Bray (editore Syngress): tiny.cc/DFamX
Hacker Journal nr. 188 - "Protezione totale"
Hacker Journal nr. 194 - "Probing & Penetration testing"
Hacker Journal nr. 197 - "Introduzione ad OpenBSD"

PROSSIMAMENTE

Esauriti i necessari richiami teorici affrontati durante il corso dell'articolo, vedremo come si configura il vero e proprio processo di difesa autonomo accennato nel medesimo.ì
Focalizzeremo pertanto la nostra attenzione sulla modalità "Active response" dell'applicativo, spostando i margini di applicazione dello stesso da Intrusion Detection System (IDS) ad Intrusion Prevention System (IPS). In quest'ottica risulterà obbligatorio rispolverare alcune nozioni già introdotte nel corso dei numeri passati della rivista relative a PF, vedendo come integrarlo al meccanismo A.R. di OSSEC.

Sarà inoltre messa sotto i riflettori anche la comoda interfaccia web dell'applicativo e l'integrazione del meccanismo di alerting a database MySQL: sicuramente un modo molto più comodo di accedere alle informazioni di nostro interesse rispetto quanto visto finora.

Concludendo, testeremo quanto realizzato in entrambe le parti attraverso la programmazione di attacchi mirati ai singoli host della rete utilizzando prevalentemente software Open Source.